2003年5月1日発行（毎月1回1日発行）第22巻第5号 昭和57年12月6日第三種郵便物認可

マイコン わかる！動かせる！プログラムが組める雑誌 マイコンベーシックマガジン

BASIC Magazine



5
2003 MAY
550 Yen

掲載プログラムがダウンロードできる！ http://www.basicmagazine.net/

PDAを使いこなして友人に差をつける!!

Palm vs PocketPC プログラミング対決！





創刊250号記念特集
第2弾
パソコン怒涛編

コンピュータのルーツを探る旅
若者たちはアミューズメントの世界へ
タイムトラベル1982~84

- ●Windows 意外と知らない便利なテクニック
- ●Visual Basic　オブジェクト指向の考え方
- ●デジタルの世界へようこそ～逆ポーランド記法
- ●HSPでレーダー作戦ゲームを作る
- ●ドットNETでタイピングゲーム



「あきば通」制作現場をチェック！

記録型パソコン用DVD技術を知る！
オーサリング＆ライティング

パソコン認定試験対策講座
パソコンリテラシ＆ネットワーク

編集部の学校訪問記



神奈川県清川村立宮ヶ瀬中学校



兵庫県立川西緑台高等学校



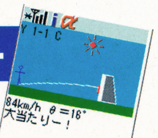
広島県立本郷工業高等学校

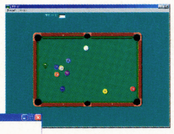


愛媛県立新居浜工業高等学校

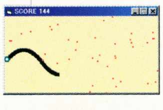
特選投稿プログラム・コーナー

VB.NET	ティ取
Visual Basic V6.0	Last is one
Visual Basic V6.0	点つぶし
Visual Basic V6.0	ビリヤード
HSP V2.55	Line Ball
HSP V2.55	3D○×
PIECE	ぷれす60 Version1.12
PIECE	UMLAUT
iアプリ	とりでの攻防
iアプリ	AFOのPK
PC-9801(N88-DiskBASIC)	バクハゲーム
ポケコン(PC-G850)	よけろ点









インテルCentrino搭載PC＆電子辞書
今月の製品情報



マイコン BASIC Magazine
http://www.basicmagazine.net

5
2003 MAY

CONTENTS

SUN	MON	TUE	WED	THU	FRI	SAT
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

巻頭カラー企画 編集部の学校訪問記

神奈川県清川村立宮ヶ瀬中学校 6

兵庫県立川西緑台高等学校 8

広島県立本郷工業高等学校 10

愛媛県立新居浜工業高等学校 12

僕たち私たちのホームページへようこそ! 14

秋葉原情報誌“あきば通”制作現場 16



創刊250号記念特集

ベーマガ編集部が行くコンピュータのルーツを探る旅 18

第2弾 パソコン怒涛編 タイムトラベル1982〜1984

若者たちはアミューズメントの世界へ

未来型“身につけるコンピュータ”「ウィービー(Weavy)」 28

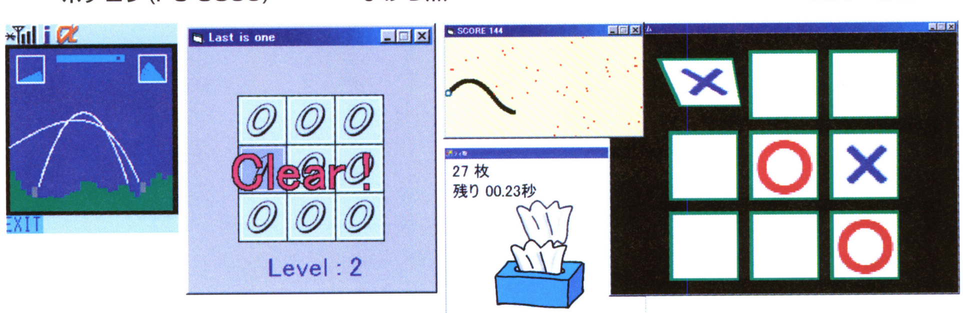
プログラム開発環境を構築しよう!

Palm vs PocketPC プログラミング対決! 30

特選投稿プログラム・コーナー 59

VB.NET	ティ取	松江 祐輔	60
Visual Basic V6.0	Last is one	井上 秀夫	65
Visual Basic V6.0	点つぶし	正凹	71
Visual Basic V6.0	ビリヤード	峠 恒司	74
HSP V2.55	Line Ball	稲津 和磨	75
HSP V2.55	3D○×	WindVoice	79
PIECE	ぷれす60 Version1.12	ていこく	81
PIECE	UMLAUT	岩崎 泰隆	86
iアプリ	とりでの攻防	shng	92
iアプリ	AFOのPK	おぐりん	97
PC-9801(N88-DiskBASIC)	バクハゲーム	吉村 直繁	99
ポケコン(PC-G850)	よけろ点	広岡 聡之	100





今月のベスト・プログラマーは誰の手に!
明日のスタープログラマー 101

掲載プログラムを改造してみる!
Dr.Dのベスト・プログラムへの道 102

© マイコンBASICマガジン5月号
第22巻 第5号 2003年4月25日印刷
2003年5月1日発行
定価 550円(本体524円)
電波新聞社 2003 Printed in Japan
編集発行人:平山哲雄
東京都品川区東五反田1-11-15
郵便番号 141-8715
電話 東京(03)3445-6111(大代表)
振替 (東京) 00150-3-51961
印刷:奥村印刷(株) 製本:(株)堅省堂

今月の製品情報
インテルCentrino搭載PC&電子辞書 36

記録型パソコン用DVD技術を知る!
第4回 オーサリング&ライティング 38

特別企画 パソコン認定試験対策講座
第7回 パソコンリテラシ&ネットワーク 115

マンガで学ぶ!コンピュータの世界!!
パソコンレクチャー ザ・ビギニング ………… 44

はじめようMicrosoft OfficeでBASICプログラミング
第13回 Windows 意外と知らない便利なテクニック ………… 48

教科「情報」など授業にも使える! Visual Basicアルゴリズム講座
第14回 オブジェクト指向の考え方 ………… 52

デジタルの世界へようこそ!
第4回 逆ポーランド記法 ………… 56

HSPdeネットワークゲーム!
第5回 レーダー作戦ゲームを作る ………… 104

.NETでゲームを作ろう!
第7回 タイピングゲーム ………… 108

オリジナルソフトが作れる!持ち歩ける!PC用携帯端末
PIECEプログラミング 第10回 ………… 123

続・ベーシックプログラミング教室
[番外編] 星音高校-BASIC基本コマンド- ………… 127

でんでんタウンロボットクラブ奮戦記③ ………… 131

読者のコーナー
あくせすROOM ………… 135

ソフト・メーカーのPRページ
SOFTWARE MAKER HOT INFORMATION ………… 136

広告索引

ソフト		**パソコンショップ**	
日本ファルコム	表3	九十九電機	表2〜前付カラー3
マイクロソフト	表4	秋月電子通商	記事中27
		パシフィックネット	後付138



清川村は，丹沢山塊に囲まれた神奈川県唯一の村です。平成10年（1998年）に完成した宮ヶ瀬ダムによって誕生した宮ヶ瀬湖を中心に広がる緑豊かな自然は，現在では観光スポットとしても知られるようになりました。

今回訪問した宮ヶ瀬中学校（荻田哲郎校長）は，この宮ヶ瀬湖畔に位置した全校生徒3人という小さな学校ですが，生徒と先生がひとつになり，少人数の良さを生かした教育が行われています。

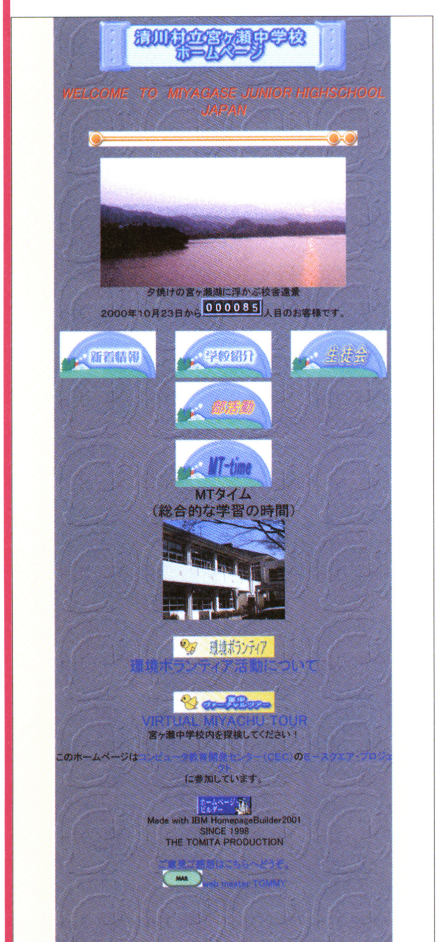
◀前列右から岡本周也くん，山本直弘くん，馬場祐昌くん，後藤裕美子先生（英語），中列右から斎藤伸之先生（数学），斎藤曜子先生（養護），後列右から中山隆司先生（理科），冨田琢哉先生（音楽），宮﨑勇弥先生（社会）

編集部の学校訪問記

愛甲郡清川村立宮ヶ瀬中学校

http://www2.ocn.ne.jp/~miyachu/

全校生徒3人が先生とひとつになって勉学に励む



▲ホームページはリニューアルされたばかり

少人数の中で豊かな人間性を育む手段となったホームページ制作

宮ヶ瀬中学校は昭和22年（1947年）に創立。宮ヶ瀬ダム建設に伴い昭和61年（1986年）に現在の地へ移転，新校舎が竣工しました。ダム建設と同時に，住民のほとんどが近隣の市町へ移転したため過疎化が進み，県内最小規模の中学校となってしまいました。

取材にうかがった3月，元気に我々取材班を迎えてくれたのは，3年生1人，1年生2人の3人の全校生徒でした。15年度も新入生1人を迎え，3人体制ということです。生徒たちは，先生とほぼマンツーマンで学習に取り組んでいます。ちなみに先生は9人在籍してい



▲やさしい眼差しで生徒を見守る荻田哲郎校長



▲案内をしていただいた冨田琢哉先生。情報教育の中心



▲宮ヶ瀬湖畔に位置する校舎は小学校と隣接している。標高300.5mのこの地区は春先でも降雪はたまにあるそうだ

ます。

同校は，平成9年から2年間，神奈川県の環境ボランティア活動推進校の指定を受け，宮ヶ瀬の自然と人とのかかわりを研究してきました。この取り組みをさらに充実させるため，平成10年，ホームページにまとめ，世界に発信しました。さらに，コンピュータ教育開発センター（CEC）の「E-スクエア・プロジェクト」にも参加するなど，ホームページによるコミュニケーションの構築は，教育目標である“豊かな人間性を育てる”一つの手段となっているようです。

ホームページは3年前から更新されていませんでしたが，今年3月に大幅リニューアルされ立ち上がりました。生



◀谷口伸一教頭が中心になって制作される「学校だより」。『一太郎』で制作



▶「MTタイム」では生徒1人に先生が2人ずつついて指導。生徒・先生が一緒に活動しているという雰囲気がうらやましい限り



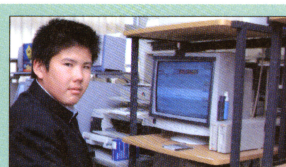
▲3年生の山本直弘くん（現・高校1年生）。「印象に残っている思い出は“炭焼き”実習かな。火止めのために夜中に学校に来ていた」と話し，卒業メモリアルビデオの編集に取り組んでいた



◀編集ソフトはユーリードの『VIDEO STUDIO 5』



▶山本くんの「MTタイム」のテーマは「川魚の調理法」だった



▲1年生の岡本周也くん（現・2年生）。インターネットで自転車に関して調べているところだった



▲1年生の馬場祐昌くん（現・2年生）。宮ヶ瀬の五つの川や湖を周り「どんな微生物がいるか」を一太郎でまとめ中



▲馬場くんと岡本くんの本発表資料

徒・先生が一つになって取り組んでいる学校生活・地域活動がさらに充実した内容で発信されています。

生徒が目的意識を持って臨む 総合的な学習の時間-MTタイム-

PC教室にはリース機のSONY「VAIO MX（MX5GK）」（Windows 2000）が2台，NECの「98Mate（Ma23D）」（Windows Me）が2台設置されています。そのほか周辺機器も充実していて，プリンタやスキャナ，デジタルカメラ／ビデオなどを活用した授業が行われています。インターネットはISDN回線を使用しています。

取材時は，「MTタイム」と呼ぶ「総合的な学習の時間」の授業を拝見しました。「MTタイム」とは，M＝みんな，で，T＝つくろうを意味しているそうです。

3年生の山本直弘くんは，「卒業メモリアルビデオ」の制作に取り組んでいました。約10時間にも及ぶ，1年間のさまざまな行事を撮影してきたビデオ映像をもとに，8分のビデオに編集します（使用ソフトはユーリードの「VIDEO STUDIO 5」）。「多くの映像から自分が映っている場面を探すのが一番苦労した」という，この「卒業メモリアルビデオ」は卒業式で披露されました。

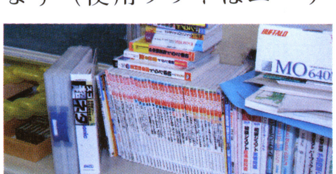
1年生の2人は，自ら決めたテーマにもとづいて，パソコンを使った課題解決学習に取り組んでいました。

岡本周也くんのテーマは「自転車の研究」です。自転車が好きだからということですが，研究内容は横回転を縦回転にする動作のしくみや，ディスクブレーキの原理・部品など，技術的なところまで調べ上げています。インターネットや文献で調べるだけでなく，同じ横→縦回転の機構を持つルアーを分解して構造を見たり，木材を使った変形自転車を組み立てたり，取り組みは多岐にわたっています。

馬場祐昌くんのテーマは，「宮ヶ瀬の川や湖に生息する微生物」です。小学校の授業で体験した水質調査に興味を持ち，引き続きそこに棲む生物を研究したいと思ったのだそうです。先生と一緒にあちこちの川に出かけ，自分の目でしっかり確かめながら研究をしてきたようすが，まとめ資料からうかがえました。

じつは，「MTタイム」の本発表が昨年12月に，保護者や宮ヶ瀬小学校の児童も参加して行われました。そのときの研究成果は紙にまとめたものでした。取り組みの最終段階として，パソコンを利用し，まとめることで，アプリケーションソフトの使い方も同時に学んでいます。使用しているソフトは「一太郎」でした。

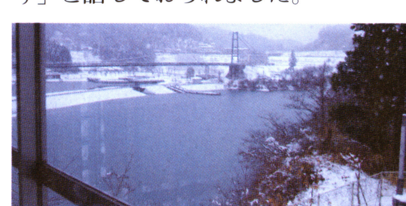
同校の情報教育の中心でもある冨田琢哉先生は「MTタイムは3年目の取り組みです。基本的に生徒がやりたいことを尊重して取り組ませています。先生方もそれに対応できる体制にあります」と話しておられました。



▲PC教室に並べられているベーマガのバックナンバー。かつてはPC部があり，プログラムを勉強する生徒もいたそうだ



▲陶芸用粘土を使って校歌を制作。取材時に3番が完成



▲校舎内から見た雪化粧の宮ヶ瀬湖

宮ヶ瀬中学校　〒243-0111 神奈川県愛甲郡清川村宮ヶ瀬954-1　☎046-288-1354

（記事中の学年表記は，2003年3月時のものです）



▲野外観察同好会のみなさん



▲図書委員のみなさん



▲放送部のみなさん

編集部の学校訪問記



▲1年生のみなさん（英語の授業で）

兵庫県立川西緑台高等学校（角倉克彦校長）は，昭和44年（1969年）に開校，今年度（平成15年度）で創立35周年を迎えます。川西市は大阪・池田市や兵庫・宝塚市などに隣接した，大阪や神戸を通勤圏とするベッドタウン。昭和30年代から始まった，「（多田）グリーンハイツ」周辺の人口増加に伴って設立された高校です。

同校は，京都大学への進学者を毎年輩出する進学校として知られる一方，平成2年には選抜高校野球大会に出場するなど「文武両道」を実現してきました。しかし最近は，全校共通の悩みである「部活離れ」も深刻のようです。

今回は，「部員不足」に悩みつつも，他校にはない魅力を発信する野外観察同好会，放送部，図書委員会を訪問しました。

兵庫県立川西緑台高等学校

http://www.hyogo-c.ed.jp/~midoridai-hs/

生徒も率先して取り組むホームページ制作。校外学習先からリアルタイム発信も！



▲コンテンツが充実しているホームページ（平成12年開設）。「みどり台のみどり」や「図書館のページ」，「英語版」は生徒が制作。「中学生向け」コーナーもあり，同校に関して親切に説明してある。またメールマガジンが月1回配信され，保護者を中心に好評を博している

メールマガジンも好評 生徒も制作するホームページ

取材時に案内してくださったのは，英語科教諭であり進路指導部員であり，情報管理部長でもある，長澤和弥先生。先生は同校ホームページの制作・管理を担当します。2月20日には同ホームページのアクセス件数が10万件を超えました。「サイトを公開して約2年半。10万という数字に責任を感じつつ，サイト訪問者である生徒，保護者，卒業生，受験生に感謝したいと思います。今後ともサイトを通じて有益な情報を提供できるよう，さらに努力していきたいと思います」とコメントしています。

とくに，修学旅行や研修旅行などの生徒たちの様子をリアルタイムで知らせる「現地レポート」は好評で，行事の際は保護者からのアクセスや掲示板の書き込みも多いそうです。修学旅行の現地レポートは過去3回実施しており，引率の先生が写真をメールで送信し，長澤先生がその内容をホームページにアップするという方法で掲載しました。

平成12年には，オーストラリア研修旅行の様子を長澤先生自身が現地からホームページを毎日更新し，保護者などに向けて発信しました。この時は研修旅行期間中だけで，千数百件ものアクセス件数を記録したということです。



▲案内をしてくれた長澤和弥先生。ホームページ制作の中心的存在

▼甲子園出場経験もある





▲川西緑台高校の外観

「みどり台のみどり」のページ制作に励む「野外観察同好会」

川西緑台高校は，その名のとおり緑豊かな自然に恵まれた環境にあります。952名（平成14年度）の生徒は，総面積65,574㎡の広大な敷地でのびのびと学んでいます。

長年にわたり，その自然の美しさ，本校敷地内やその周辺の植物（や動物など）の観察・記録とデータの蓄積を行っているのが野外観察同好会です。現在は，長年理科教員たちが校舎敷地内やその周辺の植物や動物などを観察・記録してきた膨大なデータをホームページ上に転載する作業に取り組んでいます。

部員は2年生の岡本章裕君，上村篤志君の2名だけですが，後輩に受け継ぐという，同校の歴史上重要な役目を任されています。

また，最近では2か月に一度，近隣河川の水質調査も行っており，デジタルカメラで撮影した画像とともに調査結果をホームページに掲載しています。「生活廃水や畑からの肥料の流入が川の酸性化を促し，水生の動物が減っている。もっと身の回りの水に関心を」とホームページ上で訴えています。



◀校内生物図鑑「みどり台のみどり」は過去の蓄積資料が満載



▶現在はホームページで膨大な資料を発信している

全蔵書データベース化や貸出・返却業務のコンピュータ化を担当する「図書委員会」

図書委員会も同様に，蔵書のデータベース化に取り組んでいます。最近ようやく3万冊以上の題名や作者名の記録を終えたところ。校内LANで蔵書の有無を検索できるまでに仕上がりました。今後は出版社名からの検索や貸出中か否かという情報まで，システム化したいということです。

また，図書委員会のホームページには，図書委員による日記や「緑高リレー創作（通称:アイノベル）」が掲載されています。リレー創作とは，一つの作品を複数の人がリレー形式で創作する読者参加型のもの。図書委員をはじめとする生徒たちのみによる作品です。





▲パソコンを使って貸出・返却管理をする図書委員会。図書館の3台のパソコンは生徒に開放され，進路調査やインターネット閲覧などで活用されている

◀新規開設された図書館のホームページは生徒も制作

パソコンを使ってビデオ編集に取り組む「放送部」

放送部は，部員は4名と少ないのですが，明るいキャラクターで圧倒的な存在感を見せています。

過去にはNHK杯全国高校放送コンテストでの最優秀賞やNHK会長賞を受賞するなどの実績をあげており，同校を代表する部活といえます。

アナウンス部門，朗読部門での活躍が目立ちますが，平成2年に野球部が甲子園出場を果たしたことをきっかけに，映像部門へも進出しました。部室の棚にはドキュメント制作などで撮りためたビデオがずらり。現在はSONYの「VAIO」を使用しデジタル編集を行っています。「昔は良い機材を持っている学校が強かったが，最近はどの学校もデジタル化に対応しているため，平等になった」（顧問の先生談）そうです。

機械の性能より個人の実力が試されるようになった今，4人の部員たちは，全国大会出場という目標に挑みます。



▶放送部の華々しい活躍ぶりが数々の表彰状からもわかる



◀放送部の活動。映像の編集はVAIOを使用

特徴ある教育課程



▲1年生の英語では，外国人の先生と日本人教師のチームティーチング授業が行われていた

情報教育教室



平成12年度に完成。13年度から，本格的な授業などでの使用が始まった。同校には合計58台のパソコンがあり，「国際理解」や「コンピューター造形」「情報基礎」などの授業で使用されている

川西緑台高等学校　〒666-0115 兵庫県川西市向陽台1-8　☎072-793-0361

（記事中の学年表記は，2003年3月時のものです）



▲情報技術科3年生（課題研究発表会で）

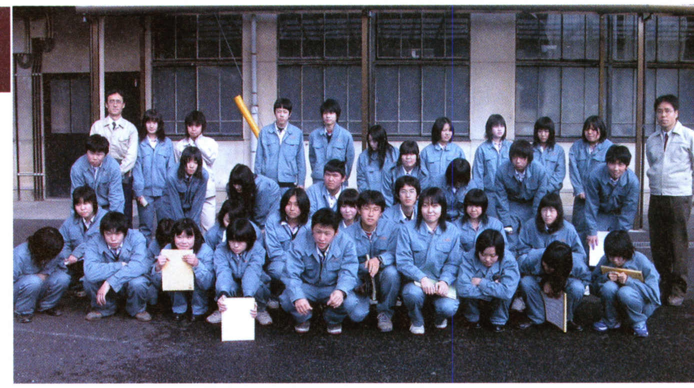


▲情報技術科2年生

編集部の学校訪問記



◀計算技術研究部



▲情報技術科1年生

広島県立本郷工業高等学校

http://www.hongo-th.hiroshima-c.ed.jp/

少人数制導入で技術を磨く「情報技術科」と検定大会上位を目指す「計算技術研究部」

広島県立本郷工業高等学校（小中谷晃校長）は，広島空港が平成5年（1993年）に移転開港し空港都市として発展している本郷町にあります。広島県内で，もっとも駅から近い高校？と言われるように，JR山陽本線本郷駅から100メートルのところに位置します。

大正8年（1919年）に豊田郡立本郷農学校として創立され，昭和40年（1965年）に単科の工業高等学校となりました。昭和43年（1968年）に現校名となり，現在，機械科，電気科，情報技術科の3科で構成されています。



▲校舎外観



▲小中谷晃校長

コンピュータ技術の基礎と基本をしっかり学ぶ「情報技術科」

本郷工業高校には主に三つのパソコン室があります。

「プログラミング室」と「アプリケーション室」には，4年リースで一昨年導入されたDELLの「OptiPlexGX110」24台と16台がそれぞれ設置されています（OSはすべてWindows2000）。これらの部屋は，ワープロ検定の練習，インターネット，レポート作りなど，自由にやってもらおうと，朝，昼，放課後と開放されており，

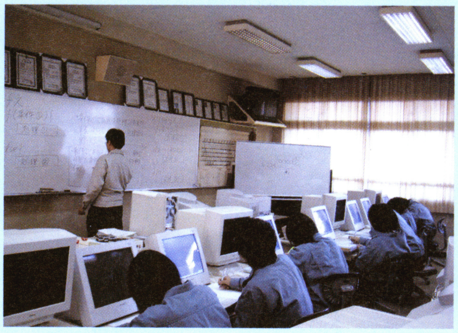


生徒が積極的に活用しています。

また「CAD室」にはNECの「PC-9821Xa20」が40台設置されています。インターネット環境は，一昨年から光ファイバーケーブルになり，校内LANも整備されているとのことです。

同校に設置されている情報技術科は，1学年1学級で，コンピュータに関するさまざまな知識や技術を中心に学んでいま

パートに分かれた少人数で学習する情報技術科 1年「工業基礎」の授業



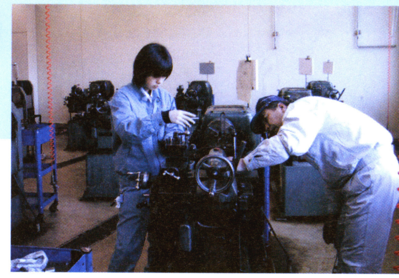
▲「プログラミング室」で行われていた1パート：C言語の授業



▲「アプリケーション室」では2パート：WORD実習が行われていた



▲こちらは3パート：シーケンス制御や電子工作授業のようす



◀4パート：機械工作の授業

す。1年生の「工業基礎」は，四つのパートに分かれていて，1年間通して授業が行われています。1パート：C言語でのプログラミング，2パート：ワードなどのアプリケーションソフトの使いかた，3パート：シーケンス制御や電子工作，4パート：機械工作で，生徒は少人数で順繰りにそれぞれのパートを学んでいきます。

基本的には，四つのパートを4人の先生が教えていますが，一つのパートを2人の先生が教えることもあるそうです。少人数なので，生徒が間近で先生のデモンストレーションを見たり，わからなければ先生が付きっきりで教えることもできるなど，生徒1人1人がじっくり確実に学ぶことができる授業です。

各人がテーマを決め グループで取り組む「課題研究」

情報技術科の3年生は，3年間の学習をもとに，1年間かけて一つのものを完成させる「課題研究」授業があります。1月22日には，その成果を発表する「2002年度課題研究発表会」が開催されました。2002年度は，興味あるテーマごとに「ランプ」「CAD設計」「ロボット相撲」「プログラミング」「画面処理」の5班に分かれて研究を重ねました。

「画面処理」は，エルサルヴァドルの工業高校の先生とスペイン語でメール交換をするなど，他にはない試みにも挑戦しました。「課題研究発表会」は，情報技術科の1年生から3年生までが参加します。1，2年生は，3年生の研究の成果，また，失敗や苦労なども参考に自分たちが3年生になったときどのように研究をやっていくかのか考えていきます。

今年度は修学旅行先が変更になり，現地の環境などもわからないことから，中止になったということですが，昨年度までは，修学旅行中に，旅行先から学校のホームページに書き込みをして，保護者などに旅先の様子を知らせるという先進的な試みも行っていたそうです。

コンテストや検定試験を目指し 日々活動する「計算技術研究部」

放課後には，計算技術研究部を訪問しました。情報技術科の天崎力くん，大下毅くんも遊びに来ていて，なごやかな雰囲気でした。しかし，時間制限を設け，一斉にカシオのポケコンを使って練習問題を解いていくときは，一転して空気が引き締まり，部員の集中力の高さを感じました。

同部の“エース”万徳栄策くんは昨年11月に行われた計算技術検定の広島県大会で，1年生ながら8位に入りました。計算技術検定は1級から4級まであり，昨年6月に4級，11月に3級を取ったそうです。同部を指導する能登正雄先生は「2年生の6月で2級，11月で1級と順調に取ってくれたら」と期待していました。万徳くんは，昨年12月に行われた校内の技術大会でも1位を取ったそうです。

今後も部員それぞれが計算技術検定の校内大会，広島県大会の上位入賞を目指してがんばっていくそうです。



▲保護者や後輩たちも見つめるなか行われた「2002年度課題研究発表会」。ロボット相撲班は，中国地区大会にも参加した

情報技術科3年「課題研究」



コンピュータ好き集まれ！「計算技術研究部」



▲カシオのポケコンを使っての練習問題に真剣に取り組む部員たち



▲1年生ながら計算技術検定広島大会8位の実績を残した万徳栄策くん（情報技術科）

本郷工業高等学校 〒729-0412広島県豊田郡本郷町本郷1443-3 ☎0848-86-2019

（記事中の学年表記は，2003年3月時のものです）

編集部の学校訪問記

愛媛県立新居浜工業高等学校（宇都宮省吾校長）は，新居浜市の中心部に位置する公立の専門高校です。前身は1937年（昭和12年）創設の愛媛県立新居農工学校で，1952年（昭和27年）に現校名に変更されました。現在は，機械科・電子機械科・電気科・電子工学科（情報電子科）工業化学科の5コースがありますが，平成15年度からは，工業化学科に変わって，環境化学科が1クラス設置されます。

全校19クラス760人が学んでいますが，情報化社会に対応するカリキュラムもしっかり組まれており，コンピュータ教育が充実している学校です。



放送部のみなさん

愛媛県立新居浜工業高等学校

http://www.ehime-net.ed.jp/

充実したPC環境のもと，生徒が参画して制作されているHP



▲放送部員が参画して更新されるホームページ。教育リンクが充実している

普通教室にも各1台 充実したPC設置環境

新居浜工業高等学校には，各普通教室に1台ずつコンピュータが設置されています。そのほか，コンピュータ室4教室にそれぞれ40台，視聴覚教室に40台，図書館に20台など，パソコンはいたるところに設置されています。合計すると，350台以上のコンピュータが完備されている同校ですが，この豊富なコンピュータを全学科の生徒が活用して授業が行われています。

インターネットを使った課題解決はもちろん，生徒が提出する書類はすべて電子化して提出したり，3年生が行う卒業課題研究論文では，生徒1人1人がPowerPointやホームページを使ってプレゼンテーション形式で行ったりしています。

一方，校内には光ファイバー網によるネットワークがはりめぐらされています。この光ファイバー網は，平成10年度と比較的早い時期に完成されました。「情報教育への取り組みは県内でもトップクラスですよ！」と同校の宇佐美東男先生。

それにともない，情報モラルの教育もしっかり行われています。「ネット社会の歩き方」を学ぶために，「学習ユニット」と「電脳商店街」という二つのソフトが用意されています。このソフトの開発には，宇佐美先生も加わっており，同校の情報教育への熱心な取り組みをうかがうことができます。生徒たちは，このソフトを使って“ネチケット”を学んでいます。



▲学校の外観



▲コンピュータ室4教室をはじめ，パソコン環境は充実

▼普通教室内にもパソコンが1台ずつ設置されている



情報モラル[ネット社会の歩き方]

二つのソフトで学習

・学習ユニットとは・・・

問題点を把握して考える学習を始めるためのアニメーション教材です。インターネットで，情報検索，情報発信，コミュニケーション，ショッピングをする際の注意事項を体験しながら学ぶことができます。

・電脳商店街とは・・・

オンラインショッピングの模擬体験を通じて，不適切な商店の見分け方を学ぶことができるソフトです。



▲学習ユニットのページ



▲電脳商店街のページ

同校は，文部省が平成6年から参加している「CEC」の情報教育実践研究指定校に毎年選ばれており，全国の学校の情報化推進を積極的に行っています。

・CECとは・・・

CECは，昭和61年7月，わが国の学校におけるコンピュータ利用促進のための基盤的技術を研究開発し，コンピュータ教育に関して普及啓発することを目的として設立された，文部科学省と経済産業省共管の財団法人です。

CECは，学識経験者，教育関係機関・団体，コンピュータ業界，ソフトウェア業界，教科書・教材業界など，広範な関係分野からの参加と，文部科学省，経済産業省，情報処理振興事業協会，日本自転車振興会などの支援を受けて活動しています。

ホームページ制作に放送部の生徒が参画

取材当日は，放送部（2年生）の活動を拝見させてもらいました。

同校では，宇佐美先生を中心として主に放送部の生徒がホームページの制作に取り組んでいます。現在の部員は2年生を中心とした9人。

運動会や文化祭など同校の活動は，“インターネット放送局”を通して紹介されていますが，この取り組みが評価され，今年の愛媛県高校放送コンテストで3位に入賞しました。過去には全国3位になったこともあり，同校の放送部の活動は以前から盛んに行われています。

[放送部部員の紹介]



・部長の平田良介くん
パソコンは中学2年生から始めました。現在は部員全員で取り組んでいる“四国美術館マップ”の制作に励んでいます！



・眞鍋正くん
新居浜市で多発している事故を調べてホームページ上にまとめたいと思っています。早く完成させたいです！



・田中英貴くん
現在自作パソコンの研究中です！



・尾﨑豊秋くん
パソコンは中学1年生から始めました！



・川上貴広くん
現在は，愛媛県の美術館を調べています。文字入力は得意です！



・石井優一くん
パソコンは中学2年生から始めています。四国の美術館マップを早く完成させたいです！

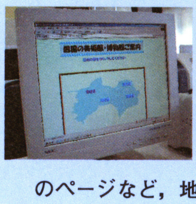


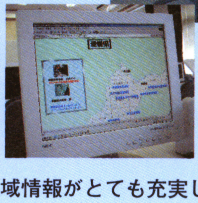
▶宇佐美東男先生。IT教育プロジェクト事務局長を務める。教育ソフトの開発にも積極的に取り組んでいる





▲放送部の活動の様子





◀放送部製作中の四国の美術館・博物館マップ。同校のホームページは，観光案内や方言のページなど，地域情報がとても充実している

新居浜工業高等学校　〒792-0004 愛媛県新居浜市北新町8-1　☎0897-37-2029

（記事中の学年表記は，2003年3月時のものです）

僕たち私たちのホームページへようこそ！

全国の学校Webを紹介

現在活躍しているスゴイ人たちも昔は子どもだった，ということで今回は今をときめく超有名人4名の小・中・高校のサイトを見てみることにしました。雲の上の人？ いえいえ，スクール・ホームページを通してみると，とっても身近な先輩に見えてきますよ。

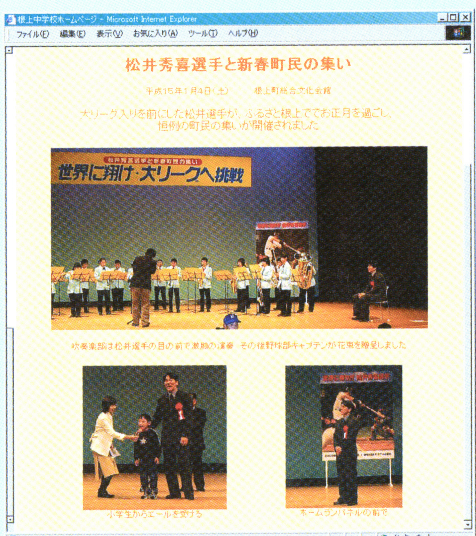
石川県根上町立根上中学校

http://www.ishikawa-c.ed.jp/˜neagaj/

今や知らない人のないメジャーリーガー・松井秀喜選手の母校です。今年のお正月を地元で過ごした松井選手の様子を取材したページが，やはり目を惹きます。松井選手の活躍を学校の後輩みんなが応援している気持ちがあふれてます。



◀光が十文字に輝いているワザアリの校舎写真です。他はテキストのみとしたシンプルな構成は見やすく，使いやすいですね。



▶全国ニュースとしても伝えられた地元交流会をどーんと特集した熱い1ページです。生徒たちが松井選手に激励を送っている場面が紹介されています。



▲韓国の中学校との交流をたくさんの写真で紹介するページです。ただ，交流のきっかけや，彼らとどんな話をしたかなどの説明文がないのは残念です。もっと文章を書こう！

桐朋中学・高等学校

http://www.toho.ed.jp/

小説家・赤川次郎氏の母校です。1日に原稿用紙80枚を書くという彼のパワーは中高の6年間を，この自由な校風のなかで過ごして培われたのかもしれません。彼の国語の成績も気になりますけどね。

▶中央写真はカーソルを乗せると別の写真に替わる「ロールオーバー」と呼ばれるしかけです。容量は重くなりますが意外な動きが出ます。





◀自分たちの日常を登校から下校までの写真で追ったページは説明文も分かりやすく，読み手を惹きつけます。

群馬県館林市立第一小学校

http://www.edu-c.pref.gunma.jp/gakko/syo/tate1es1/

宇宙飛行士・向井千秋氏の母校です。公立小学校の「ふつう」の環境から宇宙へ羽ばたく人材が巣立ったことに大きな可能性を感じます。向井さんはもともと心臓がご専門のお医者さんだったって知ってました？



◀誰もが懐かしさを感じられるすてきな校舎の写真です。右端の小さい写真は施設紹介ページへのリンクボタンです。



▶まず冒頭に「このページを見てくれてありがとう」と，ありそうで意外とないメッセージが書かれていてうれしいです。サイト運営の基本ですよね。



◀勉強は教室の中だけじゃない，ということで，かまどに薪をくべ，お釜でご飯炊きに挑戦。「昔の暮らし」を知るだけでなく，災害時などにも役立つ知恵ですね。



◀精神科医を校医として迎えるなど，紹介内容そのものもたいへん興味ぶかいです。レイアウトも代替テキスト付のクリッカブルマップやページ内ジャンプなど，構成がよくできてます。

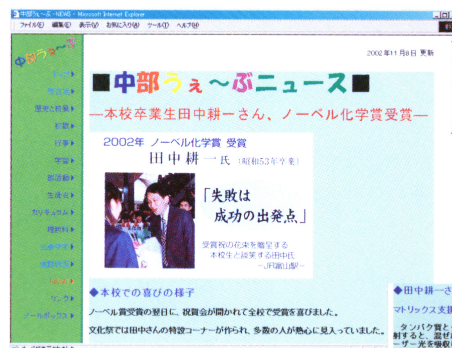
富山県立富山中部高等学校

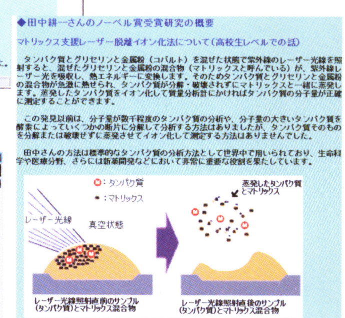
http://www2.justnet.ne.jp/~chubuks/index.htm

今もっとも有名なサラリーマン，ノーベル化学賞受賞者・田中耕一さんの母校です。このホームページの素晴らしいところは，田中さんの受賞した『マトリックス支援レーザー脱離イオン化法』という技術について高校生（とほとんどの一般人）のレベルに合わせた分かりやすい解説があることです。後輩たちによる，後輩たちのための情報発信ですね。



▶一見フレーム構造だけど，実は表（テーブル）で作成されたページレイアウトは，ユニバーサルデザインにもかなっていて，すっきり見やすい構成です。





◀▼全校をあげての受賞祝賀会の様子と，氏の研究概要を「高校生レベル」で説明した内容が書かれています。こういう分かりやすい解説を待っていた！という人も多いでしょう。



◀同校の体育大会は，4団（グループ）に分かれ3年まで同じ団に属するというユニークな伝統だとか。競技内容も独自のものが多く，毎年多数の観客で盛り上がるそうです。これほど素晴らしい特色なら，もっと内容を増やして解説してほしいところです！

特別企画 クラーク記念国際高校の生徒も参画

東京・秋葉原でしか手に入らない！

新世紀マガジン（フリーペーパー）

あきば通（つう）が出来るまで

～制作現場をチェック！～

（文／「あきば通」実行委員会）



さまざまなドラマが渦巻く，汗と涙と睡眠不足のあきば「痛」。その最新号「あきば通」第5号の制作過程を一挙大公開！

平成15年2月某日：

「あきば通」実行委員会・小林孝也（たかや）代表宛の広告掲載の問い合わせは，号を重ねるごとに確実に増えてきている。「あきば通」を取り上げたいという取材の申し込みも同様だ。着実に「あきば通」がアキバに浸透し始めていることを感じながら，彼の頭の中では次号の構想がやんわりとできあがり始めていた。



▲「あきば通」を片手にノリノリのたかや代表



▲東京・秋葉原の昼と夜

2月24日：編集会議

ある日思い立った，たかや代表が，制作スタッフを招集。

いつも眠たげ，わたなべこういち氏，人の良さがにじみ出ている沖縄出身のような中野氏，女子校の先輩タイプなtenbo様，正義感に燃えるシンヤマン君，色男であり仕事を着実にこなす（と評判の）大佐，やる気まんまん若さ溢れるクラーク高校ご一行様が，一堂に会する。

みんなでお菓子を食べながら，今回の趣旨やぶちあげなどをたかや代表より聞かされる。

3月28日発刊目標という事実に一同，顔面蒼白。すでに食べ物の味は無い。

解説しよう！3月28日発刊のためには3月17日が最終原稿入稿日である。それまでにインタビューから写真撮りから，MAP制作，さらに表紙絵も描いてもらわなくちゃいけない。



▲彼らが「あきば通」編集部の面々だ



▲笑顔でパソコンに向かう「クラーク高校」の生徒たち

・・・できるかってーの，やろう，続けようぜ！という想いがみんなの胸に渦巻くのであった・・・。

編集会議以降～3月16日（入稿前日）

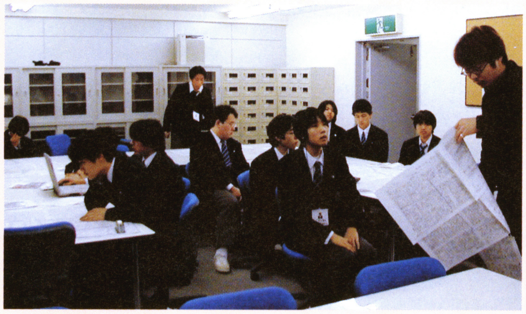
各人の分担を決めた後の連絡はだいたいがメールと電話。

あとはたかや代表がスポンサー探しに自転車を走らせる。各人，インタビューに東奔西走・・・しながらも，はらペコMAPのお気に入りの飲食店で昼食をとって一息ついていると，店主より次号はいつ出るのと声をかけられ，また，われに返り作成し続けるのであった・・・。

原稿があがってくると，わたなべ氏の一瞥で採用，不採用が判断されていく・・・・・（ちなみに不採用率95%）。

全員からの文章や写真の編集を担当するゴールキーパーのシンヤマンは，すでに仕事場に置く簡易ベッドを購入し徹夜にそなえる。可愛い奥さんのもとへと帰れる日はいつになるやら...。

そうしているうちに，刻一刻と締め切りがせまってきて，スポンサーの枠取り依頼に応対し，目尻は吊り上がり，精神的に追いつめられていく…。



▲「あきば通」の「はらペコMAP」の取材をクラーク高校の生徒に指示するわたなべ氏





▲編集作業中のシンヤマン，徹夜も辞さない



▲スポンサー探しや取材に自転車を走らせるたかや代表

地域と何かできないか！
クラーク記念国際高等学校の挑戦

「あきば通」とクラーク高校との出会いは，不思議なものであった。クラーク高校秋葉原ITキャンパス設立時の取り組みのひとつとして，タウン誌の制作を考えていたスタッフは，偶然手にした「あきば通」創刊準備号を見て感動，発行元の秋葉原周辺まちづくり研究会タウン誌編集局に電話をした。

理事長である小林たかや氏が熱く語る秋葉原の街づくりに，すっかり共鳴，その後研究会と動きを共にし，「あきば通」作成に協力を申し出た。

ところで，秋葉原でコンピュータやアニメと言えば，「おたく」を連想するが，秋葉原ITキャンパスが目指しているのは，生徒をコンピュータのマニアに育てることではなく，得意な分野の仕事を通じてコミュニケーション能力を高めること，責任感を持ち事に当たれること，などである。

「あきば通」とほぼ同時に誕生した秋葉原ITキャンパスのユニークな活動と実績が，日本の高校教育に風穴を空けていくことを期待したい。



▲クラーク高校が応援するストリートミュージシャン「空（クウ）」と生徒たち



▲ボランティア活動「お掃除志隊」



▲北米にも進出した「外神田あみたん」

3月17日入稿日

表紙絵がぎりぎりできあがってくる。今回は春らしさをアピールした，芝生に腹這いの外神田あみたんだ。今回もキミのハートを萌えさせるはずだ。



▲これが最新5号の表紙だ！

それはいいとして，締め切り30分前の段階で，インタビュー記事の推敲がまだ終わっていないこの現状をなんと説明しようか？すでに制作の最終段階担当のシンヤマンは2日目の徹夜に突入していた。

籠城用の食べ物は底を尽き，事務所の冷蔵庫にある最後の飲み物に手をつけ，オヤジギャグと記事も写真も飛び交い混迷を極めている状態。「修羅場」を垣間みる思いだ。

この後さらに，クラーク高校の皆さんが取材確認してきた飲食店情報を「あきば通」のはらペコMAPに落とすための作業が生じる。いったいどうやって間に合わせるというのだろう？かといって，最終チェックをおろそかにはできない。あきば通第3号表紙見出しの「常磐新鮮」事件（(正）常磐新線）などの二の舞は，したくない。

ボランティアの命を削って作る「あきば通」。それを手に取ってくれる皆様に是非とも立派な「あきばツウ」になってほしい。地域で60,000部の発行を誇り，さらに増刷・設置を熱望され，「あきば通」読者に快適空間を提供し続けるあきば通実行委員会。じつはボランティア精神のぶ厚いフツーの人々の集まりだった・・・。

彼らの努力がいつか実り，名実共に「あきば通」が世界の秋葉原を代表する新世紀マガジンとなることを祈ってやまない。

ベーマガ編集部が行く ～タイムトラベル1982～1984～ コンピュータのルーツを探る旅



第2弾「パソコン怒涛編：若者たちはアミューズメントの世界へ」

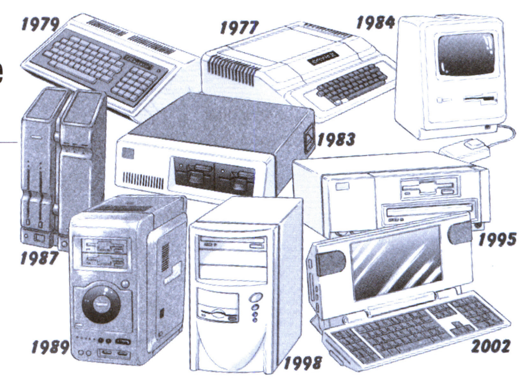
1977年(昭和52年)，東京の秋葉原に外国製のマイコン（当時は，パソコンとは言いませんでした）が襲来し，子供たちに「スッゲー，カッコイイ，ほしいよ～ん」と言わしめたのでした。しかし，日本のメーカーも黙ってはいませんでした。手先の器用さでは，我々日本人にかなう者などいないぞとばかりに，日立，シャープ，NECから相次いでマイコンが発売になったのです。

翌年の1978年，日本には，まるで神風のような「インベーダーゲーム」のブームが巻き起こります。そして，これをきっかけに国内は「マイコンブーム」へと向かっていくことになります。その火付け役となったのが，1979年に発売されたNECのマイコン「PC-8001」でした。

1980年に入ると，出遅れていたメーカーもマイコンの発売に熱をいれはじめ，1981年には個性的なマイコンも登場するなどして，これから始まるマイコン市場争奪戦に名乗りを上げたメーカーがスタートラインに出揃いました。

と，ここまでが前号のあらすじです。そして，いよいよここからが今回お話しする「パソコン怒涛編」の始まりです。どうぞ，ごゆっくりお楽しみください。

1982年 マイコンBASIC Magazineが遂に創刊！！



編集長：お待たせしました。ベーマガ編集部が行くコンピュータのルーツを探る旅の第2弾は，熱きマイコンブームの風が吹き荒れるなかで，いよいよ我らが「マイコンBASIC Magazine」が登場する1982年からのスタートです。ここから1984年までの3年間に起こったことは，まるで天に昇っていくドラゴンを見るような，あっと驚いている間に日本中にパソコンのブームが拡大していくといった，なかなか体験できないような数年を迎えるのでした。

つぐ美：前回までがパソコンの誕生にまつわるお話しで，今回がパソコンが大きく成長して日本全国に広がっていくお話しですね。今日ではIT（インフォメーション・テクノロジ）なんていってるけど，その根が広まりだしたのって，ずいぶん昔のことなんですね。

Dr.D：今となっては遠い過去のできごとじゃが，パソコンに興味を持っている読者のみんなには，ぜひぜひその頃のことを知っておいてもらいたいな。

この年，1982年は我々ベーマガ編集部にとっては独立のときを迎える記念すべき年になりました。そう，月刊「ラジオの製作」の別冊付録から，晴れて月刊「マイコンBASIC Magazine」が誕生することになった年なのです。

影：そういえば，編くんも私も，もとはラ製のキャラだったんだよね。いまじゃ，すっかりベーマガの顔になってるけど。えっ，ラ製ってなんだって？困ったなー，ラ製も知らないの？　ラセイ！

つぐ美：それいうなら，「ダセイ」でしょ。

影：月刊「ラジオの製作」は，1955年に電波新聞社から創刊されたエレクトロニクスホビーのための雑誌だったのさ。分かる？「だったのさ」ってことは過去のこと。つまり，今はなくなっちゃったけどね。編くん，いつからだっけラ製からお呼びが掛からなくなったのは。

編：さあ，どっちかというと，もとからラ製からもマイコンの編集部からも嫌われてたような気がしますけど，影さんの場合は。

Dr.D：はは，まあ，そういうな。影くんのおかげで，ラ製も大人気だったんだか



1982年フラッシュ

社会	「ホテルニュージャパン火災惨事」「日航機羽田沖墜落事故」「東北新幹線，上越新幹線開業」「500円硬貨」「日本コンピューター・グラフィックス'82 」
ヒット商品	「コンパクトディスク」「テレホンカード」「レンタルビデオソフト」「ディスクカメラ」「フラットテレビ」
流行語	「ひょうきん」「やったね」「ネクラ」「逆噴射」「ほとんどビョーキ」「ガンバリマッシュ」
流行歌	「待つわ」「セーラー服と機関銃」「北酒場」「チャコの海岸物語」「悪女」「ハイティーン★ブギ」
映画	「ブッシュマン」「ロッキー3」「レイダース・失われたアーク」「炎のランナー」「セーラー服と機関銃」「機動戦士ガンダムⅢ」「蒲田行進曲」
マンガ	「タッチ」「みゆき」「いっしょうけんめいハジメくん」「キックオフ」「パタリロ」

らな。それに，今となっては語り草になるが，今日のエレクトロニクス大国を築き上げた日本のエレクトロニクス技術者は，みんなラ製を読んで育ったわけなんじゃ。同じように，キミたちもエレクトロニクスの世界からコンピュータの世界へ移って，こうして読者のみなさんのお役に立っているわけなんだから。

つぐ美：ということは，影さんと編さんのキャラってラ製で生まれたわけだけど，私だけはベーマガオリジナルのキャラなのね。つまり，私こそ純粋なベーマガ育ちのミス・ベーマガなのネ！。

影：ミスっていったって，1人しかいないだろう。それに，力だけは男勝りだしな。

つぐ美：プン，プン！！

I/O（工学社）	1976年11月
ASCII（アスキー）	1977年7月号
マイコン（電波新聞社）	1977年9月号
RAM（廣済堂出版）	1978年2月号
LOGIN（アスキー）	1982年5月創刊号
Oh! MZ（日本ソフトバンク）	1982年6月号
Oh! PC（日本ソフトバンク）	1982年6月号
マイコンBASIC Magazine（電波新聞社）	1982年7月号
テクノポリス（徳間書店）	1982年8月号
POPCOM（小学館）	1983年5月
コンプティーク（角川書店）	1983年8月

Dr.D：ちょっと，これを見てごらん。これは，ベーマガと同じ頃に創刊されたパソコン雑誌の一覧だ。1982年以前の4誌はマイコン4大誌としてよく知られとったが，ベーマガと同じようにそのほかの雑誌はほとんどが1982年と1983年にかけて創刊されておる。これが何を意味するかは，分かるだろう。

影：分かる。ようするに，1982年からパソコン誌を読む人が増えたってことだよね。

Dr.D：そのとおり。ま，誰でも分かることだがな。いずれにしても，1982年と1983年だけで，7誌も創刊されてるわけだ。なかには，大手の出版社もまじっておることからも，当時のパソコンの普及の勢いがよく分かるじゃろう。

つぐ美：すごいのね，それだけライバル誌があったなかで，ベーマガって独走してたのね。さすが．．．。

編・影：さすが，なに？？

つぐ美：さすが，編集長だわ。こんな野蛮な連中を操って，そんな快挙を成し遂げるなんて。

編・影：シュン。

それもこれも，みなベーマガを愛読し続けてくださった，みなさんのお陰なのです。あえて言わせていただけば，ベーマガはその他の雑誌とはまったく異なる独自の路線を歩んでいました。ひたすら，読者の作ったプログラムを掲載する。そう，読者参加の雑誌だったのです。そしてもう一つ，自分で手を加えて遊ぶことの楽しさも忘れてなかったことを加えておきます。



▲1982年7月号。これが創刊号だ！





▲▶ラジオの製作付録時代と創刊前々号

各社の看板パソコン，いっせいに出揃う

この1982年には，前年に比べて2倍近くになる20数台ものパソコンが発売になりました。そのなかには，日立ベーシックマスターL3mk2やシャープMZ700，MZ1200，MZ2000などの，すでに発売されている機種のバージョンアップされた製品も含まれています。

しかし，注目すべきは先行する日立，シャープ，NECのパソコンに負けじと，その他のメーカーから登場してきたパソコンです。そして，このときに出揃ったパソコンが，その後の10年近くを互いのライバルとして競合し続けることになるのです。

6809搭載の富士通FM-7

富士通は，1981年にFM-8というFMシリーズ初代のパソコンを発売し，その1年後にこのFM-7を発売しました。CPUはFM-8に搭載された6809を，その後の1989年に80386を搭載したFM TOWNSが登場するまで採用し続けました。そして，この間の5年間で，FM-8，7，11，77，77AV，77AV20，77AV40といった名機へと育て上げていきました。

Dr.D：初代のFM-8は，がっしりしたボディにモトローラの6809のCPUを搭載しておった。当時，ザイログのZ80が人気を誇るなかで，なんとFM-8にはオプションでZ80に対応していたんだな。やはり，PC-8001のような高機能でマニアックな印象のパソコンだった。

編：それが，1年後には価格も218,000円から126,000円に値を下げて登場したんですね。機能的にはFM-8の流れを受け継いで，本体カラーもクリームと白に，さらにPSG 3重和音のサウンド機能を搭載して，多くのファンを獲得しました。

影：ところで，FMってなんの意味なの。

Dr.D：富士通マイクロつまりFujitsu Microの略なのさ。



富士通FM7

CPU：	MBL68B09（4.9/8MHz）
RAM：	64Kバイト＋VRAM 48Kバイト
ROM：	44Kバイト
価格：	126,000円



東芝パソピア

CPU：	Z80A（4MHz）
RAM：	64Kバイト＋VRAM16Kバイト
ROM：	32Kバイト
価格：	163,000円

東芝パソピア

大手の家電メーカーとして，松下に次いで東芝が発売してきたパソコンがパソピアです。パソピアって名前が，とても印象的でしたね。

編：たしか，パソコンとユートピアからの造語だったと思います。もしも間違いだったら，ごめんなさい。
Dr.D：東芝はNECのTK-80の翌年になる1977年にはTLCS-12A，TLCS-EX12/5，さらに1978年にはEX-80というマイコンキットを発売してるんだな。だから，後発のようでいて，実際には実績のあるパソコンを提供していたというわけだ。
編：たしかに，はじめて登場したときの派手さは感じませんでしたが，逆にプログラムでバグってどうのこうのというのも，あまり聞きませんでしたね。とてもおとなしくて，育ちのいいパソコンっていう印象でした。
Dr.D：その後，パソピアシリーズとしては，1983年にMSXパソコンも出しながら，それよりも数か月早くPASOPIA7という製品を発売している。これが，PSGを２個搭載してサウンド機能を強化し，価格も119,800円となってファミリー層に近い製品へと仕上がっていったな。

シャープMZ-2000

お待たせしました，シャープのMZ-2000です。MZ-80Kで大勢のファンを掴んだシャープが，次なる製品として発表したのが1981年のMZ-80Bという製品でした。このMZ-2000は80Bの廉価版に当たります。

つぐ美：ねえねえ，Dr.。なんでMZ-80Bの紹介をするんですか。これって，1981年に発売になったパソコンでしょ。
Dr.D：まあ，そういうな。前号でのお約束でもあるし，それにこのあとに紹介するシャープのX1の前に，ぜひとも紹介しておかないと出番がなくなっちまうしな。
影：Dr.って，ほんと正直だからね。1か月前のことなど，読者も忘れてるよね，編くん。
編：先週の火曜日に貸したお金，返してくれる。
影：ほらね，１週間前のことだって忘れちゃうんだもんね。
つぐ美・編：影さんだけ！！
Dr.D：ウォッホン，えー，そのMZ-80Bなんだが，じつはMZ-80Kのライバル機だったPC-8001や，その他のパソコンが登場してきたことで，MZ-80Kの弱点ともなっていたグラフィック機能を補った製品として登場したんじゃ。ただし，ちょっと値段が高すぎたな。
編：そうですね，278,000円ですからね。それでも，投稿本数はけっこうありましたよ。やはり，根強いMZファンが80Bでも増殖してたんですね。MZ-80Kとの互換性は，ほとんどなかったんですけどね。
Dr.D：というわけで，その翌年に出たのがこのMZ-2000というわけだ。80Bではオプションだったカラーグラフィックにも対応した。しかし，それでも価格は218,000円だった。それで，その後1983年に128,000円でMZ-2200を発売したんだ。ただし，モニタのない本体だけでな。
影：でも，これじゃ一体型のMZのイメージもだいなしだね。
Dr.D：やはり，時代の流れというか，その代わりシャープは新しいパソコンを誕生させるんだな。これが，X1という製品さ。



新発想のテレビパソコン シャープX1

シャープがMZシリーズの後継として発売したのが，なんとテレビパソコンのX1です。テレビとパソコンを一体化，いやテレビにパソコンを内蔵して，スーパーインポーズといったAV機能が使えるというのがセールスポイントの製品でした。さすが，目の付けどころが違いますね。

影：なつかしーな。よく，マイコンルームでテレビ見てたもんね。
つぐ美：つーことは，仕事してなかったってこと？
影：X1の投稿プログラムをチェックするんだから，しょうがなかったの。



MZ-2000

CPU：	Z80A（4MHz）
RAM：	64Kバイト＋キャラクタVRAM 2Kバイト＋グラフィックVRAM 48Kバイト
ROM：	BOOT 2Kバイト，CG ROM 2Kバイト
価格：	218,000円



X1

CPU：Z80A（4MHz）
RAM：64Kバイト＋テキストV-RAM 4Kバイト＋キャラジェネRAM 6Kバイト
ROM：IPL 4Kバイト＋キャラジェネ 2Kバイト
価格：155,000円

編：そういえば，昼休みに編集部にいないなと思うと，テレビをつけたまま居眠りしてましたよね。
影：そう，ポカポカと日差しが暖かくてつい。なんて，言わせるなよ。でもさ，X1って，本当にテレビとパソコンの画面を重ねて見ることができたんだよ。
Dr.D：当時は，テレビにパソコンの画面を表示させるというのが当たり前だったからな，いっそテレビとくっつけてというオールインワン的発想は，ここにも生きておったな。常に，先進性を失わないところに，シャープのパソコンのおもしろさがあった。
つぐ美：そうなんです。このX1をきっかけに，このあとAV機能を搭載したパソコンが登場してくるんですね。1984年に発売の，NECのPC-6601SRとか。
Dr.D：おそらく，どこのメーカーも同じ頃に同じような発想をしていたんだと思う。ただ，それが本当に製品化されるかどうかの違いでね。だから，人気の商品が出ればすぐに似たようなものが出てきても不思議ではないんだ。
編：この発想が人気を呼んで，X1はその後1987年までの間に，なんと11機種も登場するんです。もちろん，1987年には16ビット機が全盛ですから，あとは16ビット機のX68000へバトンタッチしていったということですね。



ぴゅう太

CPU：TMS9995（16ビット）
RAM：16Kバイト
ROM：20Kバイト
価格：59,800円

トミー　ぴゅう太

玩具メーカーからもパソコンが発売されるようになり，トミーのぴゅう太はなんと日本語Gベーシックという，日本語で入力ができるベーシック言語を搭載して，話題を呼びました。しかも，16ビットマシンです。

Dr.D：簡単に動画がプログラムできるという楽しさがあったな。
影：「イケ」「トマレ」「オワリ」って感じだよね。
編：「イケ」が「GOTO」で「トマレ」が「STOP」，「オワリ」は「END」でしょ。なんか，英語がしゃべれる人ってプログラムも簡単にできそうな気がするな。
つぐ美：今も日本語BASICがあったら，おもしろいと思うわ。ほら，アイボとかアシモといったロボットが生活に入ってきてるでしょ。これを日本語のプログラムで動かせたら，いいと思わない。
影：「マワレ」「カエレ」「デーブ」
つぐ美：「マワレ」が「FOR」で，「カエレ」が「RETURN」ね。えっ，「デーブ」ってなによ。
影：SAVE。

ビジネスパソコンの元祖 NEC PC-9801

玩具メーカーからの参入がある一方で，パソコンは本来のビジネス利用へと動き出します。その第一歩を記したパソコンが，NECのPC-9801です。PC-8001がマイコンとして登場してから，わずか3年後のことです。いうまでもなく，このPC-9801は日本のビジネスパソコンの怪物として急成長を遂げることになります。

Dr.D：初代のPC-9801は，おせじにもビジネスで使えるシロモノではなかったな。なにせ，標準でフロッピーディスクも付いてなかったわけだからな。それに，外付けのフロッピーディスクは，8インチだからな。データの保存は，もちろんカセットテープだったんじゃ。しかし，16ビット機の能力には，さすがに驚かされたわい。当時は，東芝のパソピア16とか，松下のマイブレーン3000とかが16ビット機であったがな。
編：で，どのくらい速かったんですか。
Dr.D：1から10000の数字を表示させるのに，8ビット機のPC-8801が3分08秒かかったのが，PC-9801はたったの55秒だった。
影：たしかに，驚かされたよね。でも，今から想像するとテープ版とかフロッピーディスク版のビジネスソフトって，とても使えそうにないよね。
Dr.D：それもそうじゃが，そういう我々もPC-9801をPC-8001と同じゲーム機として見とったからな。ハッハッハー。
つぐ美：みなさん，とても平和だった時代のベーマガ編集部ののどかな雰囲気ががお分かりいただけましたでしょうか。



NECPC-9801

CPU：μPD8086(5MHz)
RAM：128Kバイト（最大640Kバイト）
ROM：N88-BASIC(86)，モニタ96Kバイト
価格：298,000円

1983年 夏の花火のように各社のパソコンが咲き乱れ

しかし，それでもなお，この年には驚くような事件が二つ起こったのです。と，そのことについては後回しにして，こうして増殖するパソコンとベーマガの読者の数に，ベーマガ編集部もちょっとあわてていました。

というのも，ベーマガ創刊号の1982年7月号で掲載したプログラムの数は15機種36本でしたが，それが1983年1月号には23機種43本に増えていました。しかし，まだそれだけで驚くのは早いわけで，この年の12月号には，なんと47機種67本もの数のプログラムを掲載するまでになっていたのです。わずか1年半の間に，じつに掲載機種は32機種も増えていたことになるのです。

編：これって，喜ぶべきことですよね，影さん。
影：ああ，本当はね。でもさ，はっきり言ってさ，ものすごかったよな。
編・影：朝から晩までマイコンルームに閉じこもったまま，読者のプログラムを1本ずつチェックしたんだもんネー。

ちなみに，当時の投稿プログラム本数は，月に1000本を越えていたのです。しかも，ベーマガの表紙には毎号のように新しいパソコンの名前が載っていました。

そして，この年の5月25日から28日までの3日間，東京は平和島の流通センターで「マイクロコンピュータショー'83」が開催されました。

編：しっかり覚えてますよ。我々も電波新聞のブースに出かけたんでしたヨネ。そしたら，会場に大勢のベーマガ読者が来てくれて，サインなんか迫られちゃって困っちゃた。
影：そういう感じじゃなかったな。喜んでというか，人をつかまえてはサイン書いてたじゃん。

この会場で話題をさらったパソコンは，東芝パソピア7/5，シャープMZ-2200，三菱MULTI8など10機種ほどありました。

このように，年に一度のマイコンショーが開かれるたびに，ベーマガの読者はもちろん全国からパソコンファンが，この会場につめかけました。そして，こうした光景がこのあともしばらく続くのです。

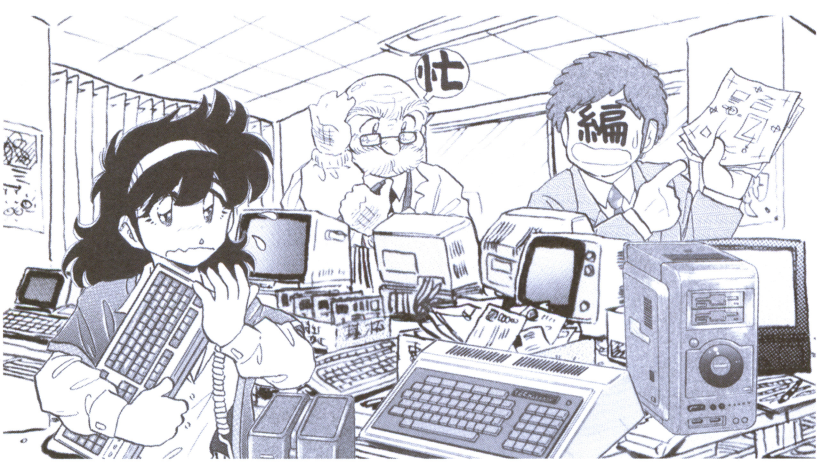
国産規格のMSXパソコンが登場

さて，1983年に起こった二つの事件のうち一つは，MSXパソコンの登場でした。MSXパソコンとは，パソコンの統一規格として誕生したもので，この年にホームユースに向けたパソコンのMSX規格が設けられ，この規格に基づいてメーカー各社がこぞってMSXパソコンの発売を始めたのです。MSXとは，この規格を提唱したMicrosoftの名前にあやかってMicrosoft-Xから付けられたそうです。

考えようによっては，先行するNECとシャープ，富士通のパソコンとの格差を縮めるべく大手の家電メーカーが共同で発売したパソコンといえなくもありません。当時は，台数的にもファンの数からしても上位の3社がずば抜けていたからです。

MSX規格のパソコンであれば，どこのメーカーの製品でも同じプログラムが動きます。こうして6月に規格がまとまり，ベーマガの11月号には松下のMSXパソコンCF-2000の発表記事とともに，2本の「バイオリズム」（きむらくーん）と「FIGHTER」（ウイングマン）というプログラムが掲載されました。

編：すばやい対応，てんこ盛りのプログラム本数が，なんといってもベーマガの特徴でしたからね。とにかく，新製品が出たらプログラムを載せろーって，編集長がうるさく言ってたよね。
影：それなら給料も乗せろーって言った





1983年フラッシュ

社会	「おしんブーム」「東京ディズニーランド開園」「シンデレラコンプレックスシンドローム」「グラビア誌FOCUS」「大韓航空機撃墜事件」
ヒット商品	「パーソナル無線」「インバーターエアコン」「ワンルームマンション」「MSXホームパソコン」「パソコンテレビ」「男性用育毛剤」「無印良品」
流行語	「いいとも！」「○○の輪ッ！」「積木くずし」「おじさんは怒ってるんだぞ」
流行歌	「さざんかの宿」「矢切りの渡し」「君に胸キュン」「めだかの兄妹」「時をかける少女」「悲しい色やね」
映画	「E.T.」「スターウォーズ・ジェダイの復讐」「フラッシュダンス」「ランボー」「愛と青春の旅だち」「戦場のメリークリスマス」「南極物語」「探偵物語」「時をかける少女」
マンガ	「童夢」「めぞん一刻」「あした天気になあれ」「バツ＆テリー」「気まぐれコンセプト」



◀ベーマガ裏表紙を飾り続けた工藤夕貴とHI

日立H1

CPU：	Z-80A
RAM：	32Kバイト
ROM：	40Kバイト
価格：	62,800円

ら，編集長なんて言ったと思う。
つぐ美：なーに？
影：お前には，はじめから乗せる給料がないだってさ。シュン。
編・つぐ美：おお，すばらしき真実！！なにせ，影だからね。パチパチパチ．．．。

こうして，この年には年末にかけてサンヨーMPC-10，ヤマハYIS-503，東芝パソピアIQ，富士通FM-X，キヤノンV-10といったMSXパソコンが相次いで発売になりました。
その後，MSXの規格は1985年にグラフィック機能を強化したMSX2，さらには1988年には日本語機能を強化したMSX2+，そして最後には16ビット機対応のMSX TurboRへと発展していきました。

Dr.D：とにかく，カートリッジを差すだけでゲームやミュージックなどのゲームがすぐ楽しめたり，サウンドやグラフィックスの機能もよくできていて親しみやすい設計になっていたな。
編：当時は，人気タレントを組み合わせたCMも多かったね。
影：東芝のパソピアは，横山やすし，岡田有希子だろ。ソニーのHITBITが松田聖子。そして，松下がキングコングだ。
編：キングコングもタレントなの？　でも，やっぱり印象に残ってるのは，ベーマガの裏表紙をずっと飾ってくれた日立H1の工藤夕貴ちゃんじゃないかな。

ファミリーコンピューターの登場

MSXの発表と時を同じくして，この年の7月に発売されたのが，あの任天堂のファミリーコンピューター(ファミコン)でした。登場したての頃はさほど人気もなかったファミコンですが，それが予想をはるかに超える大ブームを巻き起こすきっかけを作ったのは，1985年に発売されたスーパーマリオブラザーズの大ヒットでした。

編：今もよく覚えているんですが，五反田の東急デパートの玩具売り場にファミコンを見に行ったんです。そしたら，あまりに玩具って感じで，それまでにもテーブルテニスのようなテレビゲームってありましたよね。だから，第一印象はほとんどなかったんですよね。
Dr.D：それは，無理もないな。言ってみれば，ファミコンはカートリッジが重要であって，本体にはさほどの魅力もないわけだから。パソコンのように搭載するCPUがどうのこうのって世界でもないしな。
影：それでも，ドンキーコングとか，それなりのゲームもすでにあったんだよね。
Dr.D：しかし，一つのゲームソフトのヒットが，ゲーム機本体を一躍メジャーに仕立て上げてしまうのだから，ソフトビジネスというのは分からんもんじゃな。
影：なに，そのソフトクリームとかビジネスホテルだとかって。
Dr.D：ほれ，電波新聞でもMZシリーズやPC-8000シリーズのゲームを販売してたことがあっただろう。デンパマイコンソフトといって。あのことさ。
影：うんうん，あったよね。カセットテープでさ，ちょうどマイコンルームのあった電波ビル別館の1階でね。たまたま通りかかると，すぐに手伝わされるんだよな。体育会系の販売部長がいてさ。それが，どうしたの？
Dr.D：なんとも。つまり，1本でも優れたソフトが出ると，その本体の人気もぐんと上がるということさ。ファミコンが，それをみごとに証明したってことだ。もっとも，その後もヒットが続かなければ苦しいが，当時はファミコンにはライバルが存在してなかったからな。だから，続々とソフトメーカーが参入して，ドラゴンクエストなどの人気ゲームを作り上げていった。
編：そのファミコンも，1990年にスーパーファミコンが発売になったことで一応の終わりを告げることになります。でも，その後も1994年にNEWファミコンが発売になるまでいろろなゲームが出てましたから，しばらくは人気が続いていましたね。
Dr.D：そういえば，ファミリーベーシックというのもあったな。キーボードとカートリッジがセットになった。たしか，ベーマガでも取り上げたことがあったな。メモリが2Kバイトほどしかないから，たいしたゲームもできないんだが，それでも当時のベーシックを組みたい子供たちには魅力的なオプションだったな。
編：データの保存は，パソコンと同じようにカセットテープでしたね。
影：ここで最新情報だけど，ファミコンの初期型のコントローラはボタンが四角だったんだけど，その後は丸いボタンになったんだって。
編：どこが，最新情報ですか。誰だって，とっくに知ってますよ。

別冊スーパーSoftマガジンでゲーム攻略へ

こうした変化の激しい1983年，ベーマガにも同じような変化が見られました。それが，11月号から付録として登場したスーパーSoftマガジンです。
時代は，1978年のインベーダーゲームのブーム以降，少しずつではありましたが，若者のビデオゲームへの関心が高まりつつありました。また，ベーマガの読者からみても，自らプログラムを組んで作るゲームの最高峰に位置するアミューズメントの世界に関心を持つのは当然のことだったのかもしれません。
そして，11月号の付録で編集長（私）は「ソフトウェアを新たな文化媒体としてとらえ，本格的に取り組んだ誌面を提



ファミコン（写真はファミリーベーシックをセットしたところ）

CPU：	N-CHカスタムチップ（6502互換）
RAM：	2K×8ビット
ROM：	最大32K×8ビット
価格：	14,800円

供する目的で発行していきます」と宣言したのです。

影：スーパーSoftマガジンの第１回目の特集は，うる星あんず氏による「目指せ100万点！ MAPPY大解析」だね。言わずと知れた，ナムコのMAPPYのゲーム攻略解析さ。これ，私もはまったね。
つぐ美：ますます，影さんの公私のけじめがなくなっていったわけね。
影：だって，20ページにわたって，こと細かに解説されてたら，それをじっくりと検証するのが編集部員の役目でしょ。そりゃあ，2，3日の徹夜も覚悟でしたよ。ホント。
つぐ美：それで，変臭部員かって，古いギャクを飛ばさせないでよ！！
影:ノリノリついでに，さらに12月号の特集を言っちゃうと，これが「めざせ1000万点XEVIOUS大解析PART1」なんだよね。これまた15ページだよ。しかも，PART1ってことは，次もあるってこと。実際には，PART3まであったんだから。もう，10日間は，マイコンルームで野宿体制ですよ。ホント。
つぐ美：どうにもしようのない影さんだけど，その単純さがベーマガ読者のバロメータでもあるのよね。それに，ほんとのこと言って，私もゲームにははまっちゃったからネ。これって，ミイラ取りがミイラになるって言うのよ。
影：それを言うなら，ミイラ取りが墓にはまって，あまりに体が太いんで出られなかった…だろ。



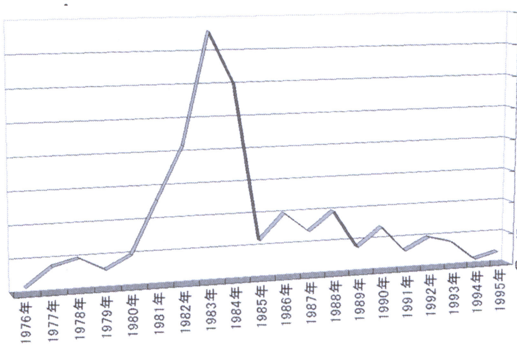
▲1983.11月号に付録として初登場のスーパーソフトマガジン

こうして，1983年の年末は，マイコンルームが突如としてゲームセンターに変身してしまったのでした。



▶その後，スーパーソフトマガジンは歴史を塗り替えるほどの勢いを見せ，本誌にとじ込まれるまでの1984.12月号まで続いた

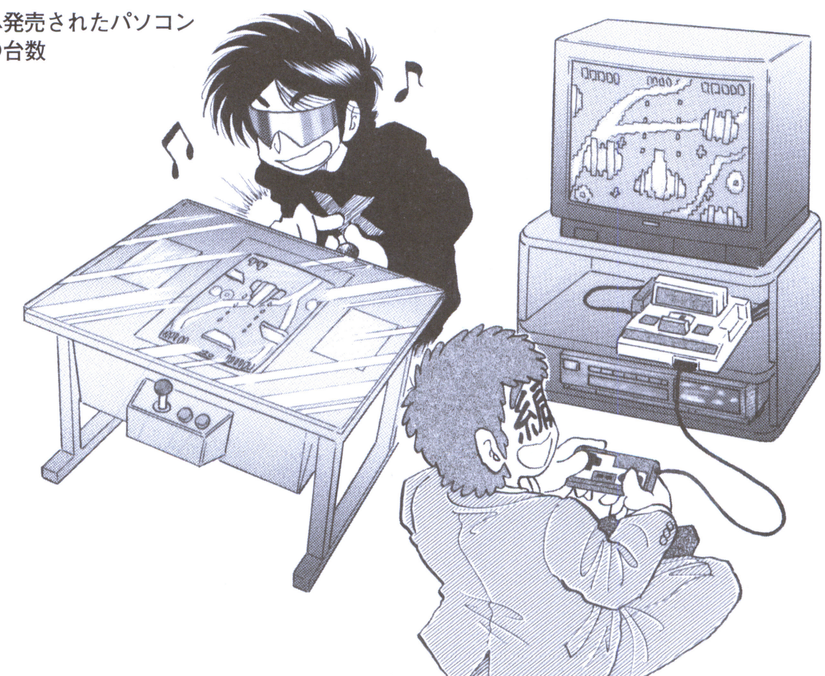
1984年 若者たちはアミューズメントの世界へ



▲発売されたパソコンの台数

1983年1月号で23機種43本だったベーマガのプログラム掲載本数も，1984年1月号では48機種64本に，さらに12月号には53機種70本という本数を記録し，プログラム掲載本数でもいよいよピークを迎えたのでした。

もうここまでくると，発売されるパソコンもみな互いに似かよった機能を搭載するようになり，しだいしだいに一つの



1984年フラッシュ

社会	「ロサンゼルスオリンピック」「グリコ・森永事件」「日本初の衛星放送」「ニューメディア」「バイク宅急便」
ヒット商品	「おいしい水」「男性用化粧品」「ポケット型カラー液晶テレビ」「缶入りウーロン茶」「キャベツ人形」「環境ビデオ」
流行語	「飽食の時代」「くれない族」「マル金　マルビ」「イッキ・イッキ」
流行歌	「スリラー」「ワインレッドの心」「悲しくてジェラシー」「涙のリクエスト」「娘よ」「桃色吐息」
映画	「インディジョーンズ魔宮の伝説」「フットルース」「さよならジュピター」「風の谷のナウシカ」「お葬式」「瀬戸内少年野球団」
マンガ	「六三四の剣」「バリバリ伝説」「天地を喰らう」「アキラ」「まんだら屋の良太」「課長　島耕作」

パソコンのあるべき姿へ近づいていくような感じがしていました。

編：1983年と1984年は，ものすごい勢いでパソコンが出てきましたよね。いちいちどのパソコンの新機種なんだか，確認するのも面倒なくらいよく登場してきました。

Dr.D：そうだろうな。ほれ，前ページのグラフを見てごらん。これは，前回の特集の第一弾で紹介した1976年から1995年までの間に発売されたパソコンの台数をグラフで表したものなんじゃが。見てのとおり，1980年から急上昇して1983年にピークを迎えていることがわかる。そして，同じような変化で1984年から1985年には台数が減っている。

つぐ美：ずいぶん激しい変化なんですね。これって，やっぱりファミコンとかMSXの登場が影響してるのかしら。

Dr.D：うむ，いろいろな見方があるじゃろうが，ワシが思うにはホビーとビジネスの二つにパソコンユーザーが分裂しはじめたのが，この年なのではないかと。それと，なんといってもファミコンの登場が大きい。1983年に出た頃には，さほど注目するほどのものでもなかったが，1985年に登場したスーパーマリオブラザーズの大ヒットで，いっきに子供たちの目がファミコンに向いてしまった。

編：そうすると，ファミリー層を狙っていたMSXも，出鼻をくじかれたようなかたちになってしまったというわけですね。

Dr.D：そうだな。それと，もう一つには子供たちの玩具として手に入れられる価格の範囲に，パソコンが値を下げられなかったことも影響してるだろうな。なにせ，ファミコンは1万円台で手に入るわけだから。しかも，操作が簡単で誰でもすぐにゲームに夢中になれる。難しい知識や操作を知らなくていいというのは，プログラムを組んでゲームを作るというのとはまったく正反対だからな。

編：そうはいっても，ベーマガの読者も火がついたように活発にプログラムを投稿するようになりました。お陰で，この年には平均して65本以上のプログラムを掲載するほどでしたから。

●この時期のパソコン一覧 ※カッコ内の数字は発売月を表わしています

西暦	国内のパソコン	海外のパソコン	OS
1982	エプソン HC20（ハンドヘルド）		
	NEC N5200/05	IBM PC/XT発表	
	NEC PC-9801（10）（元祖98）	IBM PCjr（11）	
	NEC PC-2001（ハンドヘルド）	Commodore Commodore64	
	富士通 FM7（11）	Commodore MAX MACHINE	
	富士通 FM11シリーズ		
	シャープ MZ700		
	シャープ MZ2000		
	シャープ MZ1200		
	シャープ X1		
	ソニー SMC70（11）（元祖3.5FD）		
	日立 ベーシックマスターL3mk2		
	日立 MB16001		
	東芝 パソピア		
	松下 MY BRAIN3000		
	松下 JR200		
	カシオ FP1000／1100		
	NEC PC-8001mk II		
	トミー ぴゅう太		
	三洋 PHC-25		
	三洋 PHC-8000（ハンドヘルド）		
1983	NEC PC-8201（ハンドヘルド）	Apple社 Lisa（5）	
	NEC PC-9801F1/F2（FD搭載98）	IBM PCポータブル（2）	
	ソード M5		
	MSX規格誕生（10）		
	NEC PC-8801mk II（11）		
	NEC PC-6001mk II（6）（しゃべるパソコン）		
	NEC PC-6601（11）		
	NEC PC-6601mk II		
	NEC PC-9801E（11）		
	NEC PC100		
	NEC PC-8201		
	シャープ MZ5500（10）		
	シャープ MZ2200（7）		
	シャープ X1（12）		
	シャープ X1C/D		
	シャープ PC5000		
	カシオ FP-200（ハンドヘルド）		
	カシオ FP3000		
	沖電気 IF800/50		
	日立 ベーシックマスターL3mk5（5）		
	日立 H1（12）（MSX）		
	東芝 パソピア5		
	東芝 パソピア16		
	東芝 パソピア7		
	東芝 パソピアIQ（11）（ワープロソフト内蔵MSX）		
	東芝 パソピアmini（ハンドヘルド）		
	三菱 MULTI8		
	松下電器 JR800（ハンドヘルド）（B5サイズ）		
	松下電器 JR300（パソコンテレビ）		
	三洋 MBC-55		
	三洋 WAVY10（12）（MSX）		
	ソード M68		
	ソニー SMC777		
	キヤノン X-07		
	ヤマハ YIS503/303（11）（MSX）		
	セガ SC-3000		
	任天堂 ファミリーコンピュータ（7）		
	バンダイ RX-78		
1984	富士通 FM77（5）(3.5FD搭載)	Apple社 Macintosh（1）	MacOS
	富士通 FM NEW7（5）	IBM PC/AT	
	富士通 FM-X（MSX規格対応）	Apple社 Apple II C（4）	
	NEC PC-9801F3		
	日本IBM IBM PC/JX		
	NEC PC-9801M2発表（11）		
	富士通 FM16 β発表		
	NEC PC-6601SR（11）（Mr.PC/六本木パソコン）		
	NEC PC-6001mk II SR（11）		
	シャープ MZ1500（5）		
	シャープ MZ6500		
	シャープ X1turbo（12）		
	シャープ PC-1245（ポケコン）		
	エプソン HC88/80（ハンドヘルド）		
	エプソン HC40/41（ハンドヘルド）		
	富士通 FM11BS（2）		
	富士通 FM11AD2		
	日立 B16/EX		
	日立 S1（5）		
	日立 S1モデル40		
	三洋 MBC55-2		
	三洋 MBC6800		
	キヤノン AS300		
	ソニー HitBit HB-75/55(3)		
	カシオ PV-7		
	松下 CF-2000		
	松下 CF-2700		
	三菱 ML-F120D		
	三菱 ML-F110		
	三菱 ML-F120		

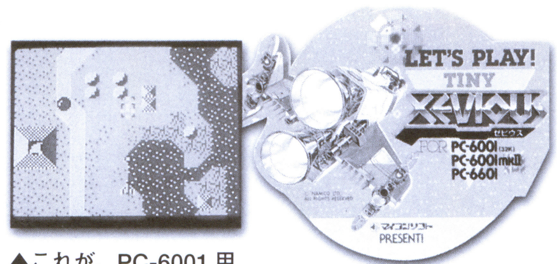
別冊「ALL ABOUT NAMCO」の発刊

結局，前年からスタートした別冊付録のスーパーSoftマガジンは，この1984年の1年間は別冊のまま掲載を続けることになりました。

つぐ美：やっぱり，ナムコのXEVIOUSは最高だったわね。それに，マイコンソフトでもPC-6001シリーズ用の「TINY XEVIOUS」やX1用の「XEVIOUS for X1」を発売しましたしね。

特集

1984年1月号	「めざせ1000万点 XEVIOUS 大解析 PART2」
1984年2月号	「めざせ1000万点 XEVIOUS 大解析 PART3」
1984年3月号	「LibbleRabble 大解析」
1984年4月号	「機動戦士ガンダム1」
1984年5月号	「ミコとアケミのジャングルアドベンチャー」
1984年6月号	「ギャプラス」
1984年7月号	「XEVIOUS for X1」
1984年8月号	「サラダの国のトマト姫」
1984年9月号	「ドルアーガの塔」
1984年10月号	「パックランド」
1984年11月号	「サザンクロス前編」
1984年12月号	「サザンクロス後編」



▲これが，PC-6001用『TINY ゼビウス』だ

影：あともう一つ，スーパーSoftマガジンで忘れちゃいけないのが，「CHALLENGE HIGH SCORE」のコーナーだよね。全国のゲームセンターでたたき出されたハイスコアが，ランキングされて掲載されるというやつさ。

編：これもまた，読者参加のイベントですね。こうなると，ベーマガはゲーム一色。これは，ゲーマーにとってもゲームプログラマーを目指す若者にとっても，ベーマガは手放せなくなっちゃうよね。

影：そうだろう。日本ゲーム推進協議会の名誉顧問としては，非常にうれしい言葉を言ってくれた。いやいや，久しぶりに君のことを見直したよ。その意気，その意気ネ！

つぐ美：それと山下章氏の「チャレンジアドベンチャーゲーム」とか，「ペーパーアドベンチャー」のコーナーも人気だったわ。

そして，このスーパーSoftマガジンも翌年の1985年からはベーマガの本体に吸収されて発刊されることになったのです。

と，ここまでビデオゲームの話を進めたら，最後に行き着くところはスーパーSoftマガジンのなかからナムコのゲーム情報を1冊の本にまとめた「ALL



▲オールアバウトナムコは，今でも語り継がれている（現在は絶版）

ベーマガだって，28万部を達成したよ!!

この時期，急上昇したのはベーマガの掲載機種やプログラムの本数だけではありませんでした。そう，ベーマガの読者数もまたこの数年間であれよあれよという間に増え続けていったのです。

82年7月号の創刊時の発行部数は9万5千部だったのが，83年1月号には11万5千部，同7月号18万部と急上昇を続けたのです。そして，84年2月号では遂に20万部を突破し，どこまで記録が更新されるやらハラハラドキドキで...。

というわけで，創刊からまる3年の85年7月号で，発行部数25万5千部を達成することになったのです（ベーマガ史上最高は28万3千部）。当時の読者のみなさん，あらためてお礼申し上げます。ありがとうございました。

Dr.D：このときの勢いには，さすがのワシもびっくりじゃった。

影：サーフィンしてたら，その勢いで山にまで登っちゃったって感じだったね。

編・つぐ美：そのたとえ，ウマイ！！まさにビッグウェーブ。99点！！



ABOUT NAMCO」シリーズでしょう。

当時，ゲームの最高峰はなんといっても「ゼビウス」に代表されるナムコの一連のゲーム。その内容も以下のようになっていました。

目次

第1章・ ナムコゲームの魅力をさぐる
第2章・ 世界のスーパー・キャラクター パックマンとその仲間たち
第3章・ ナムコスペース・ゲーム大研究
第4章・ ナムコキャラクター・ゲーム研究
第5章・ ナムコニュータイプ・ゲーム研究
第6章・ パソコンで楽しめるナムコゲーム
第7章・ ナムコグッズ総カタログ
第8章・ サウンドで楽しむナムコゲーム ナムコミュージック全集
第9章・ ナムコキャラクター総図鑑

今となっては絶版ですが，全422ページのこの本は，ゲームの世界をめざす人たちのバイブル的存在でもあったようです。



コンピュータとともにベーマガは歩む

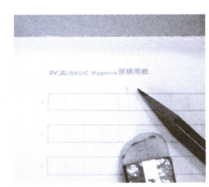
第2弾「パソコン怒涛編」，いかがでしたでしょうか。前回を含めても，10年足らずの間に日本のコンピュータ事情に大きな変化が訪れたことがお分かりいただけたかと思います。ただし，これはあくまでもベーマガ読者のレベルで見た場合の話で，当時はまだ現在のように一般の人までもがパソコンに関心をもっていたわけではありません。そうなるまでには，それからさらに10年の月日が経過することになります。

さて，今日のコンピュータ社会のなかで，かつてはベーマガの読者だったというプログラマーやエンジニアの方がたくさんいらっしゃいます。この20年の歩みのなかで，当時子供だった人たちが成長を遂げて今の社会を支えています。ベーマガの役割は，こうした多感期の子供たちに，ほんの少しのきっかけを与えたに過ぎません。なぜならば，当時の若者にとって，自分の家にコンピュータがあること自体が夢だったからであり，その夢を手にしたときの感動をずっと忘れていなかったからです。

21世紀は，アトムの時代。コンピュータ社会への夢が，少しずつ現実のものとなりつつあります。そして今，日本は物を作る時代からサービスの時代へ，ハードからソフトの時代へと移りつつあります。この変化に対応するには，やはりコンピュータの力が必要になります。

今回の特集でお伝えしたかったことは，こうしたビッグバンのような変化が，20年前に起きていたという事実です。さらに，これまで道のなかったところへわずかな道ができると，そこに大勢の人が集まり，さらに新しい道がその先にできていきます。こうした動きがコンピュータの世界で起こる限り，その道標としてベーマガもみなさんといっしょにその先の道を歩んでみたいと思います。

Dr.D：ならば，そろそろ参ろうかな。
影:ちょっと，Dr.。どうしちゃったの急に。
Dr.D:編集長も言ってるじゃないか，道を求めて旅をしなければならんのじゃよ。
つぐ美:はー，それって水戸黄門さんね。
影：つーことは，編くんが助さんで私が角さん？？
Dr.D：では，ここで一句。”みちゆけば，かわにおよんでおおはしこえる，みなみのくもにつぐみかな”。ハッハッハッ...
影:おいおい，ちょっと待ってよ～！Drがついに壊れちまったよ。どうしよう。
つぐ美：ということは，私はあのくの一の，えーと，ねえ，ちょっと待って！
影：おーい，編ク～ン，早くしないと迷子になっちゃうゾー！
編：それでは，みなさん。またお会いしましょう。バイ！！



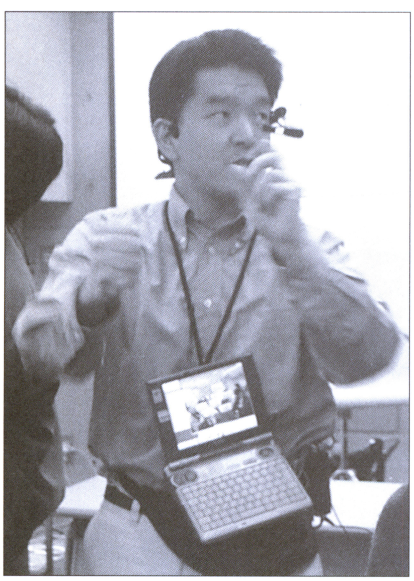
発見，未来のコンピュータ！

“身につけるコンピュータ”「ウィービー（Weavy）」

独立行政法人 産業技術総合研究所
http://www.aist.go.jp/
知能システム研究部門「ウィービー」のWEBサイト
http://www.is.aist.go.jp/weavy/
取材・文／松原拓也

ノートPCやPDAなど，携帯性のある電子機器が普及している昨今ですが，「独立行政法人　産業技術総合研究所（茨城県つくば市）」の知能システム研究部門では，それをさらに推し進めた，「ウェアラブルPC（身につけるコンピュータ）」の研究が行われています。

ウェアラブル・ビジュアル・インターフェース「ウィービー（Weavy）」



▲「ウィービー」を実演する研究メンバーの蔵田武志さん※

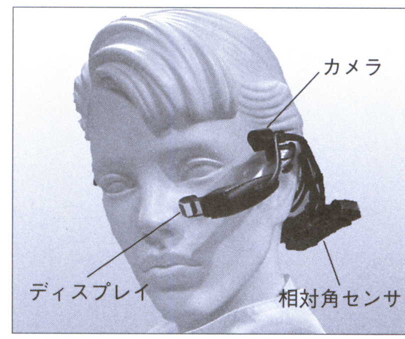


▲「ハンドジェスチャインターフェース」のデモ。「つまむ」動作がマウスのクリックに相当する※

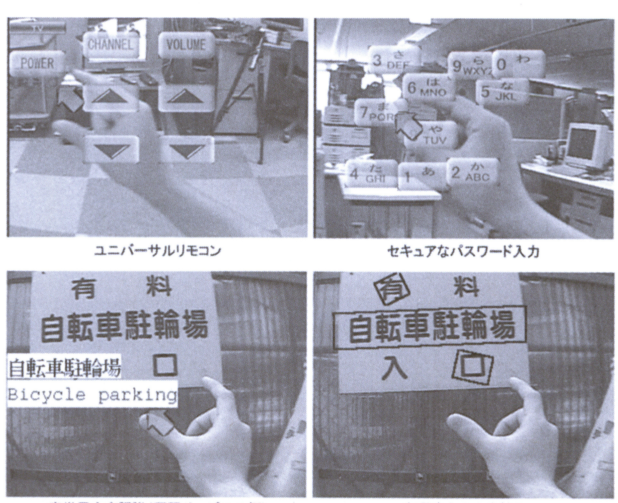
蔵田武志さん，大隈隆史さん，興梠正克さん，加藤丈和さんらが行っているのは，「ウィービー（Weavy）」と名づけられた，ウェアラブル・ビジュアル・インターフェース（入出力装置）の研究です。

そして，「ウィービー」を備えた着用型のシステムが「ワイバーン（Wyvern）」です。「ワイバーン」は，頭部に取り付けるカメラとディスプレイなどが一体になったヘッドセット型の装置，胴体に取り付けるセンサ類，そしてウェアラブルPCとで構成されています。

ユーザーは，このシステムによりコンピュータからさまざまな支援を受けることができます。



▲頭部に装着するカメラとディスプレイ，相対角センサ（加速度センサ，ジャイロなど）



▲「ハンドジェスチャインターフェース」の一例。手の動きだけで，PCを操作。文字認識や翻訳も自在＊

◆「ハンドジェスチャインターフェース」

ユーザーの手の動きを認識するという機能です。例えば，手の動きだけでメニュー選択や文字入力が行え，さらに看板に書いてある文字を翻訳する（実世界文字認識技術）といった操作も可能です。

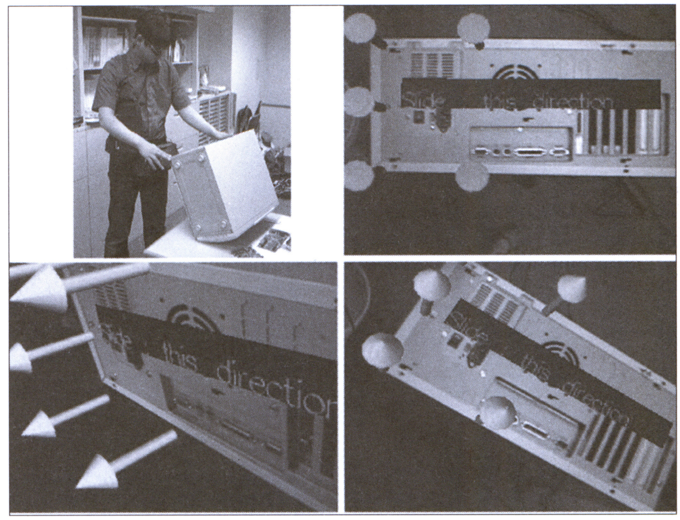
◆「パーソナルポジショニング」

ユーザーが今，どこに

※産業総合技術研究所 知能システム研究部門 研究成果展示会 ～オープンハウス2002～にて



▲「パーソナルポジショニング」の一例。ユーザーの位置と向きをリアルタイム表示。これでもう道に迷わない！＊



▲「3次元物体追跡に基づく拡張現実情報提示」の一例。ネジ穴の位置など，パソコンの組立方法をわかりやすく教えてくれる＊

いて，どの方向に向いているかを表示する機能です。位置情報は，加速度センサ，ジャイロ，磁気センサ，カメラ画像の信号から求めています。

◆「3次元物体追跡に基づく拡張現実情報提示」

実際に見ている映像に，仮想的な3次元映像を重ね合わせる機能です。この技術はオンラインマニュアル（3次元映像の説明書！）や，教育分野に応用可能とのことです。

＊資料提供：産業技術総合研究所　知能システム研究部門

顔を自動的に記憶するカメラ　ウェアラブル・アクティブカメラ(WAC)

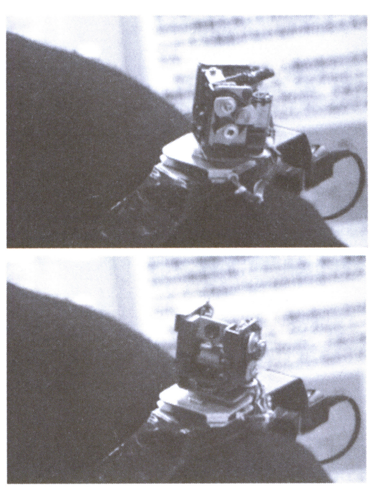
「ウェアラブル・アクティブカメラ(WAC)」は，人の顔を自動的に覚えてくれるカメラです。カメラは，ユーザーの肩の上に乗せ，そして，制御用のウェアラブルPCを腰に装着します。

カメラは，自動的に向きを変え，視界に入った人の顔画像を取り込みます。撮影した画像は無線LANを介して，外部のサーバに蓄積，データベースを構築していきます。



▲肩に乗せることがユニークな「ウェアラブル・アクティブカメラ(WAC)」。重さはわずか100g

顔の認識結果はPDAなどに出力できます（ウェアラブルPCとPDA間の通信はBluetooth）。この一連の処理は自動的に行えるので，例えば，ユーザーが意図しなくても，「この人，誰だっけ？」と思った時には，すでに認識結果を表示させることができます。



◀自動的に向きを変え、顔を撮影するカメラ

これらの技術は，エピソード（いつどこで出会ったか）記憶想起支援，ペットロボットのPIM（アドレス帳など）機能，セキュリティなどへの応用が考えられているそうです。



▲顔画像は，ウェアラブルPCと外部のサーバで分散処理されている



▲実演する研究メンバーの加藤文和さん。顔の認識結果をPDAに表示

◆SFを超えたコンピュータ！◆

コンピュータを身につけ，より自然にユーザーを支援するという未来的な発想に驚きました。「ウィービー」は目下，開発中で，今後は，研究機関向けにモニタ販売を行い，より研究を深めていく予定とのことです。実用化される日が楽しみです。

特集 無料で開発ツールをゲット！



Palm（パーム） vs PocketPC（ポケット） プログラミング対決！

Palm／PocketPC用ソフトを開発できる環境を構築しよう！

携帯情報端末（PDA）として注目を集めているPocketPC，Palm。PocketPCはカシオやコンパック，東芝など多くのメーカーが製品を発売しており，豊富なラインアップが魅力。一方のPalmは，日本ではソニーが中心となって続々と新モデルを投入。スマートなデザインと使いやすさで人気を集めています。

超コンパクトな個人向け携帯端末という位置づけである以上，ベーマガとしても黙って見過ごすわけにはいきません。そこで今回は，プログラミングという視点でこの二つに迫ってみました。

紹介している開発環境は，インターネットに接続すればすぐに入手できるものばかり。記事を読みつつ環境をととのえてぜひ挑戦してみましょう。

Palm編 今こそ挑戦！ Palmウエア・プログラミング!!

（文／松原拓也）

小さなボディに大きな可能性を秘めたPalm

「PalmOS」は1996年の登場以来，シンプルで使いやすい操作性によって多くのユーザーに親しまれている，携帯情報端末（PDA）用のプラットフォームです。今回はそんなPalm OSの環境で動作するアプリケーション（Palmウエア）の作成方法を紹介したいと思います。Palm OSを採用しているPDAには，パームコンピューティングの「Palm」シリーズ，ソニーの「CLIE（クリエ）」シリーズ，ハンドスプリングの「Visor（バイザー）」シリーズなどがあります。各機種には互換性があり，それぞれ共通のPalmウエアが動作します（上位互換）。

主なPalm製品

Palmの楽しさを凝縮したスタンダード機



『Palm m130』（実売価格22,000円前後）

メーカー	パームコンピューティング
URL	http://www.palm-japan.com/
OS	Palm OS 4.1
CPU	モトローラ Dragonball VZ 33MHz
メモリ	DRAM8Mバイト
ディスプレイ	160×160ドットカラー液晶
インターフェース	シリアル/USB/SDカード MMCスロット/赤外線通信
問い合わせ先	インフォメーションセンター japancare@corp.palm.com

国内で人気のCLIEシリーズ最新モデル



『PEG-NZ90』（実売価格80,000円前後）

メーカー	ソニー
URL	http://www.sony.jp/CLIE/
OS	Palm OS 5
CPU	Intel PXA250 200MHz
メモリ	DRAM16Mバイト
ディスプレイ	320×480ドットカラー液晶
インターフェース	USB/通信カード・スロット/メモリー・スティックスロット/AVリモコン・ポート/ヘッドホン・ステレオミニ・ジャック/赤外線通信
その他	200万画素デジタル・カメラ，FeliCaリーダー内蔵，Bluetooth対応，180度回転液晶パネル搭載
問い合わせ先	お客様ご相談センター☎0570-00-3311

Palmウェアの開発環境をそろえよう

Palmには“デベロッパー・プログラム”という開発者向けのサービスが用意されています。CLIEシリーズのユーザーなら，ソニーの「CLIE」デベロッパー・サイト（http://www.jp.sonypdadev.com/），その他のユーザーはパームソース（Palm Source,Inc.）のデベロッパー・サイト（http://www.palmos-japan.com/developers/）で，ユーザー登録（無料）を行うとよいでしょう。

次にROMイメージをダウンロードします。一部の機種は，手持ちのPalm本体からROMイメージを取得できます。最新OSであるPalm OS 5は『Palm OS 5 Simulator（シミュレータ）』でPalmウェアを実行できるので，ROMイメージは不要です。なお自

作Palmウェアを配布する場合，Creator IDという4文字のコードが必要です。Creator IDは，パームソース社のデベロッパー・サイトで登録（無料）して入手しておきましょう。

Palmウエアの開発環境として最も有名なのが，メトロワークスの『Code Warrior Development Studio for Palm OS Platform』です。同ソフトは，エディタ，C/C++コンパイラ，デバッガ，『Palm OS Constructor（リソース・エディタ）』，SDK，『Palm OS Emulator』，『Palm OS Simulator』など，Palmウェアの開発に必要なツールがすべて含まれています。なお，開発にあたっては，C/C++言語によるプログラミングの知識が必要です。

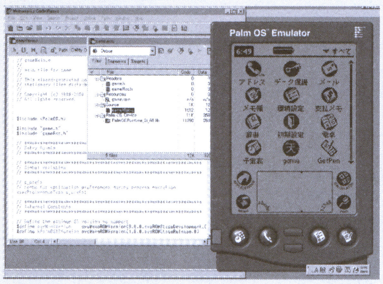
このほかに“gcc”というC/C++コンパイラを使って開発する方法があります。gccは無償で入手できるメリットがありますが，初期設定が難しいので今回は省略します。

またBASICによる開発環境を提供するキットとして，『NS BASIC/Palm』があります。簡単操作でフォームをデザインでき，『Visual Basic』などの経験を活かしたい人にはこちらがオススメです。

Palmウエアを開発できる主な環境

Palm OS 5対応の統合開発環境

『CodeWarrior Development Studio for Palm OS Platform Japanese Edition バージョン9.0 Windows版』

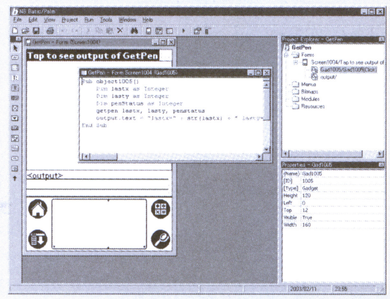


メーカー：メトロワークス
URL：http://www.metrowerks.co.jp/
価格：49,800円（消費税別）
※学生・教育関係者向けのアカデミック版（19,800円／消費税別）もあります。

▲開発ソフト『CodeWarrior for Palm OS Platform』（左）と『Palm OS Emulator』（右）を並べたところ

BASICでPalmウエアをプログラミング!!

『NS Basic/Palm』



メーカー：NS Basic Corporation
URL：http://www.nsbasic.com/palm/Japanese/
価格：$149.95
対応機種：Palm OS 3.0以上，Windows 95/98/ME/NT/2000/XP
※『NS Basic/Palm』はBASIC言語のPalm OS用開発環境です。メーカーWebサイトからデモ版（約3.5Mバイト）を無償ダウンロードできます。

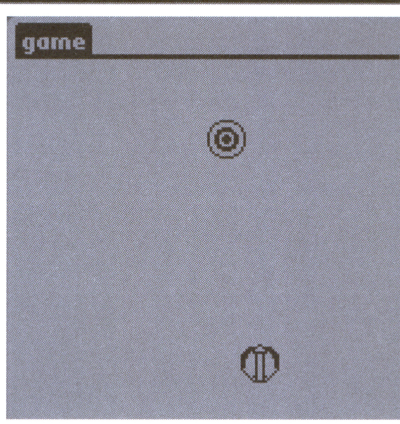
▲『NS Basic/Palm』の動作画面。オブジェクト指向で，プログラミング初心者にとっつきやすそう

Palmで動作するゲームを作ってみよう！

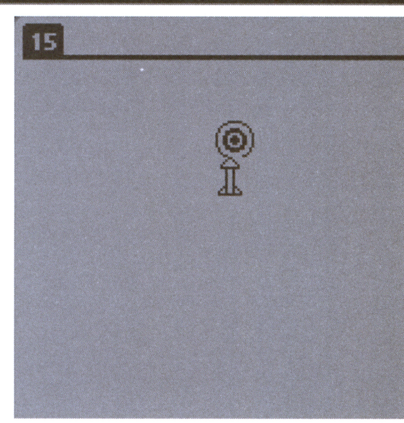
では，『CodeWarrior Development Studio for Palm OS Platform Japanese Edition バージョン9.0 Windows版』（以降CodeWarrior for Palm OSと表記）でPalm実機上で動作する，マト当てゲームをプログラミングしてみましょう。

スタイラス（ペン）で画面（Palmではフォームと呼びます）を軽くドラッグすると，その方向に矢が発射されます。当たった場所が中心に近いほど高得点。的から外れてしまうとやり直し，というのが主な処理内容です。

「マト当てゲーム for Palm OS」（Palm OS 3.5以上対応）



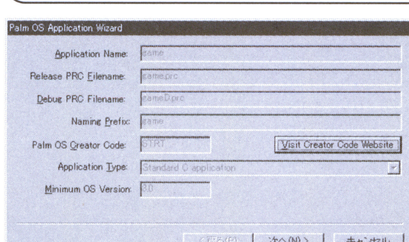
▲スタイラス（ペン）で画面をドラッグして矢の発射方向を決め，的にうまく当てよう



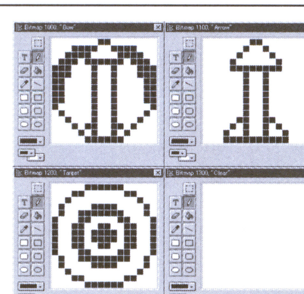
▲的に矢が当たると，画面左上に得点が表示される

主なプログラミングの流れ

CodeWarrior for Palm OSとPalm OS Emulatorを起動したらプログラム開始です。“File-New”をメニュー選択して“Palm OS Application Wizard”を実行します。“Project name”のテキスト・ボックスにプロジェクト名“game”を入力すると，プログラムのひな型が自動生成されます（**写真1**）。つぎのページにプログラムの修正箇所を紹介するので（**リスト1**），それに沿って“game Main.c”を書き替えていきましょう。



《写真1》“Palm OS Application Wizard”でプログラムを自動生成！



《写真2》ビットマップ・リソースの作成。ドット・イラストを描く感覚で進めよう

その後，“Constractor”という専用ツールを使って，“Bow（弓）”，“Clear（空白）”，“Arrow（矢）”，“Target（マト）”の各ビットマップ・リソースを作成してください（**写真2**）。

《リスト１》Palm用マト当てゲーム作成用修正差分リスト

```
//「マト当てゲーム」for Palm OS (差分ソース) by matsubara

#define WIN_XX        160        //画面サイズwidth[pixel]
#define WIN_YY        160        //画面サイズheight[pixel]
#define MATO_X        (WIN_XX/2)      //的のX座標
#define MATO_Y        (WIN_YY/4)      //的のY座標

static int        eventwait = evtWaitForever; //ウエイト間隔
static int        arrowX; //矢のX座標
static int        arrowY; //矢のY座標
static int        x1;     //矢のX移動量
static int        y1;     //矢のY移動量

//-----------------------------------------
// ビットマップ描画
//
static void DrawBitmap(int x, int y, int id)
{
    MemHandle h;      //メモリハンドル
    BitmapType* p;    //ビットマップのポインタ

    h = DmGetResource('Tbmp', id);     //リソースの取得
    if (h != NULL) {
        p = MemHandleLock(h);     //メモリハンドルのロック
        WinDrawBitmap((BitmapPtr) p, x, y); //描画
        MemHandleUnlock(h);          //メモリハンドルの開放
        DmReleaseResource(h);     //リソースの開放
    }
}

//-----------------------------------------
// 弓の引き始め状態
//
static void ChargeArrow(int x,int y)
{
    if((x1==0)&&(y1==0)&&(y>WIN_YY/2)) { //発射前なら、
        arrowX = (x - 8);     //発射座標を格納
        arrowY = (y - 8);
        DrawBitmap(arrowX, arrowY,  BowBitmap); //弓の描画
        DrawBitmap(MATO_X ,MATO_Y, TargetBitmap);//的の描画
    }
}

//-----------------------------------------
// 矢の発射
//
static void LaunchArrow(int x,int y)
{
    if((x1==0)&&(y1==0)&&(y>WIN_YY/2)) { //発射前なら、
        eventwait     = 10; //ウエイト間隔を設定
        x1 = ((x-8) - arrowX) ;     //移動量算出
        y1 = ((y-8) - arrowY) ;
        if(y1==0)
            y1=-1;     //動かない場合の安全策
    }
}

//-----------------------------------------
// 矢の移動・再描画
//
static void RedrawArrow(void)
{
    int dx,dy;
    char tmp[15];
    FormType * frmP;

    if((x1==0)&&(y1==0))
        return;

    DrawBitmap(arrowX, arrowY, ClearBitmap);  //矢の消去
    arrowX += x1;     //座標を加算
    arrowY += y1;
      DrawBitmap(arrowX, arrowY, ArrowBitmap);     //矢の描画


if((arrowY<=0)||(arrowY>=WIN_XX)||(arrowX<=0)||(arrowX>=WIN_YY
)) {
        FrmGotoForm(MainForm);//画面外処理
    }else{
        dx = Abs(MATO_X-arrowX);
        dy = Abs(MATO_Y-arrowY);
        if((dx<16)&&(dy<16)){ //的に命中、
            frmP = FrmGetActiveForm();
            StrPrintF( tmp,"%d",(16-dx)+(16-dy) );
            FrmCopyTitle( frmP,tmp ); //得点表示
            eventwait = evtWaitForever; //ウエイト間隔を無限に
            x1        = 0;
            y1        = 0;
        }
    }
}
```

■1■
DrawBitMap関数
ビットマップを画面に描画する処理を行います。

■2■
ChargeArrow関数
スタイラスで画面が押された瞬間に呼び出されます。ドラッグの開始位置を記憶します。

■3■
LaunchArrow関数
スタイラスが画面から離れた瞬間に呼び出されます。ドラッグの終了位置から，矢の移動量を求めます。

■4■
RedrawArrow関数
矢を動かす処理を行います。この関数は一定時間に呼び出されます。矢が的に当たったとき，FrmCopyTitle関数でメニュー・バーに得点を表示させます。この関数の引数には，フォームのポインタが必要です。ポインタはFrmGetActiveForm関数で取得します。

■5■
MainFormHandleEvent
Wizardで生成される関数です。メインフォームのイベントをすべてて処理しています。

```
//-----------------------------------------
// メイン画面のイベント処理
//
static Boolean MainFormHandleEvent(EventType * eventP)
{
    Boolean handled = false;
    FormType * frmP;

    switch (eventP->eType)
        {
        case menuEvent:
            return MainFormDoCommand(eventP-
>data.menu.itemID);

        case frmOpenEvent:
            frmP = FrmGetActiveForm();
            MainFormInit(frmP);
            FrmDrawForm(frmP);
            eventwait = evtWaitForever; //ウエイト間隔を無限に
            x1              = 0;
            y1              = 0;
            DrawBitmap(MATO_X ,MATO_Y, TargetBitmap);
            handled = true;
            break;

        case frmUpdateEvent:
            break;

        case penDownEvent: // スタイラスが押された
               ChargeArrow(eventP->screenX ,eventP->screenY);
            break;

        case penUpEvent: // スタイラスが離れた
            LaunchArrow(eventP->screenX ,eventP->screenY);
            break;

        case nilEvent:  // 時間によるイベント
            RedrawArrow();//矢の移動・再描画
            break;

        default:
            break;
        }
    return handled;
}
```

■6■
AppEventLoop
Wizardで生成される関数です。その際，EvtGetEventの2番目の引数を“eventwait”に書き替えてください。

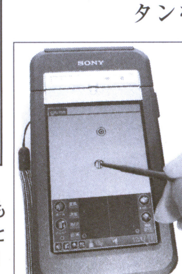
```
//-----------------------------------------
//
// アプリケーションのイベント処理
static void AppEventLoop(void)
{
    UInt16 error;
    EventType event;

    do {
        EvtGetEvent(&event, eventwait);

        if (! SysHandleEvent(&event))
        {
            if (! MenuHandleEvent(0, &event, &error))
            {
                if (! AppHandleEvent(&event))
                {
                   FrmDispatchEvent(&event);
                }
            }
        }
    } while (event.eType != appStopEvent);
}
```

コンパイルしたら エミュレータで動作チェック！

プロジェクト・ウィンドウのMakeボタンを押すとコンパイルが行われ，ファイルがビルドされます。Runボタンを押して，エミュレータ上で動作チェックしてみましょう。実機で動かしたい場合は，HotSync（ホットシンク）という機能でPalm本体に転送します。実機でゲームが動作すればプログラミングは成功です！



《写真3》いつでもどこでも遊べるのが，Palmのイイところ。ほしいゲームは作る！これがベーマガ流！

PocketPC編 PocketPCなら環境をすぐにそろえられる！

（文／山崎　寛）

PocketPCの開発環境をあなたの手に！

PocketPCは，マイクロソフトのWindows CEを搭載し，Windowsとの親和性が高いパワフルなPDAといえます。

私もつねに携帯してMP3プレーヤにしているほか，電車での移動中や待ち時間にはビデオ観賞，さらにインターネットでのちょっとした調べものなどなど，AirH"（エアエッジ）とともに手放せない存在になっています。

とても便利なPocketPCですが，「こんなソフトがあれば……」なんて思うこともあります。そんなとき，ベーマガ読者なら，自分でプログラムを組んでしまうという解決策がありますね！

そんなわけで，PocketPC用の開発環境があると何かと便利なわけですが，これがなんとマイクロソフトから無料で提供されているのです。それが『eMbedded（エンベデッド） Visual（ビジュアル） Tools（ツールズ） 3.0』（以降eVT 3.0と表記）。主な入手方法を右のコラムで紹介しているので，そちらを参考にしてください。

『eMbedded Visual Tools 3.0』とは？

Windows 98SE/Me/2000/XPで動作する，Windows CE用アプリケーション開発ツールです。提供元はマイクロソフト。『eMbedded Visual Basic 3.0』（以降eVBと表記），『eMbedded Visual C++ 3.0』（以降eVCと表記）の二つから構成されており，名前が示すとおり『Visual Basic』（以降VBと表記）や『Visual C++』（以降VCと表記）によく似た統合環境となっています。eVT 3.0登場前のWindows CE用開発環境はVBやVCのアドオン・ソフトだったため，『Visual Studio』などが必要でしたが，eVT 3.0では単体でソフト開発が可能になっています。

VBやVCと似ているがちょっと違う

入手したeVT 3.0をセットアップすると，eVBとeVCがスタート・メニューに登録されます（**写真5**）。

eVB（**写真6**），eVC（**写真7**）の基本画面を見ればわかるように，それぞれVB，VCに非常によく似ています。おおまかな流れはVB，VCそのままだと思っ

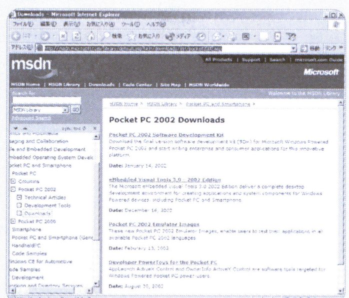
---

eMbedded Visual Tools 3.0を手に入れよう！！　手段は三つ！

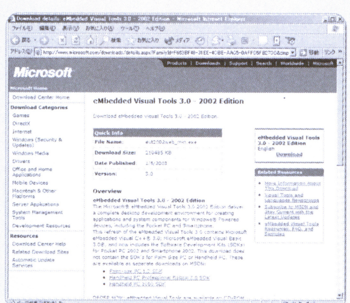
その1　英語版をマイクロソフトの英語版Webページからダウンロード！

マイクロソフトのWebページ（http://msdn.microsoft.com/library/default.asp?url=/downloads/list/pocket2002.asp）から“eMbedded Visual Tools 3.0 - 2002 Edition”の項目を選び（**写真1～写真2**），“Download”のリンクから実行ファイルダウンロードします（URLは変更になっている可能性があるので，見つからない場合は検索で“eMbedded Visual Tools”と入力して探そう）。

ダウンロードしたファイルは自己解凍形式になっていて，実行するといったんハードディスク上に展開され，その後にセットアップを開始します。なお，この英語版にはPocketPC 2002のSDKが含まれています。



《写真1》英語版eMbedded Visual Tools 3.0はWebからダウンロード可能

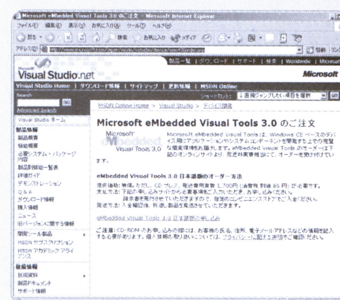


《写真2》このダウンロードには，PocketPC 2002のSDKが含まれている

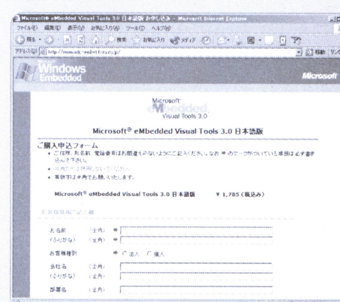
その2　日本語版CD-ROMをマイクロソフトの日本語Webサイトから注文！

同じくマイクロソフトのWebページ（http://www.microsoft.com/japan/msdn/vstudio/device/emvt30order.asp）へアクセスし，フォームに必要事項を記入して送信（**写真3～写真4**）。支払い用紙が郵送されるので，最寄りのコンビニエンス・ストアで送料と手数料計1,785円（料金は変更されることがあります）を払い込みます。マイクロソフトで支払いが確認された後，CD-ROMが手元に届けられます。

なおCD-ROM版の場合，PocketPC 2002のSDKは含まれていないようです。別途つぎのURL（http://msdn.microsoft.com/downloads/default.asp?url=/downloads/sample.asp?url=/msdn-files/027/001/908/msdncompositedoc.xml）からSDKをダウンロードしましょう。



《写真3》日本語版はCD-ROMを注文する形になる



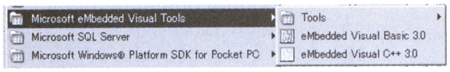
《写真4》必要事項を記入して送信しよう。数日で請求書が送られてくる

その3　雑誌／書籍の付録CD-ROMからゲット！

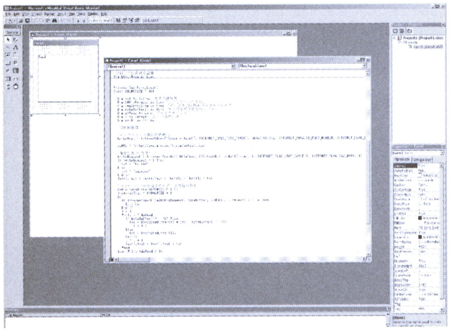
三つ目の入手法法は，付録CD-ROMなどに『eMbedded Visual Tools 3.0』が収録されている本を購入するというものです。Windows CE関連のプログラミング解説書などを探してみるといいでしょう。ただ，解説書の類はたいてい高価です。購入前に値段を見て懐具合と相談してください。

どれを使うといいの？

実際のところ，日本語版といってもメニューの文字が“File”から“ファイル”になっている程度。ヘルプなどは英語のままの箇所も多いので，ことさら日本語版にこだわる必要はないと思います。高速な通信回線があってすぐに入手したい人はダウンロードを，そうでない場合はCD-ROMの注文をするといいでしょう。



《写真5》スタートメニューに登録されたツール

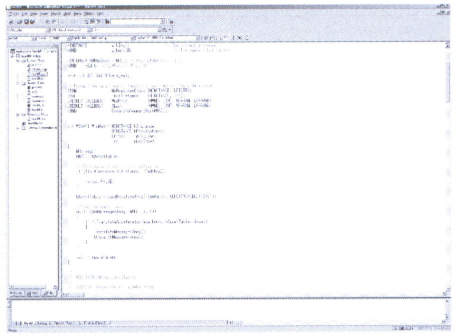


《写真6》eVBの画面。構成はVB6.0とよく似ている

ていいでしょう。eVBの場合でも，フォームにコントロールを配置し，各イベントにコードを入力，APIを宣言して使用するなど，VBとなんら変わることなくプログラミングを進められます。

作成したプログラムの実行は，SDKに含まれるエミュレータ上や（**写真8**），実際に接続したPocketPCで行います。もちろん，コンパイルしたファイルをコピーすればPocketPC単体で実行が可能です（eVBの場合，EXEファイルではなく中間コードで書かれた“～.VB”というファイルを作成することになる）。

ただeVB，eVCどちらも全体的にVB，VCの簡易版といった印象で，ステートメントや関数，APIにおいて削られているものが多数あります。そのため，VBやVCで作成したプログラムを動かそうとした場合，画面サイズなどインターフェースの問題を除いたとしても，たいていはそのままでは動作しません。呼び出すAPIや関数など，かなりの部分で作り直しを要求されるでしょう。逆に，eVBやeVCのプログラムなら，少ない手直しでVBやVCで動作できる可能性があります。



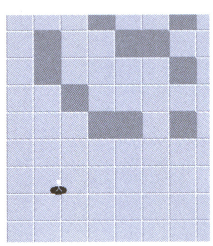
《写真7》eVCの画面。VCを使い慣れた人なら，違和感なく使えるだろう



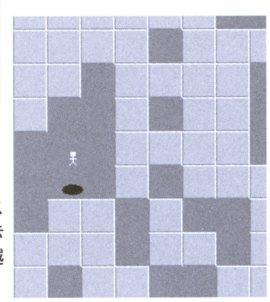
《写真8》エミュレータを使い，パソコン上で高速に動作テストやデバッグが可能

eVBでプログラミングに挑戦！

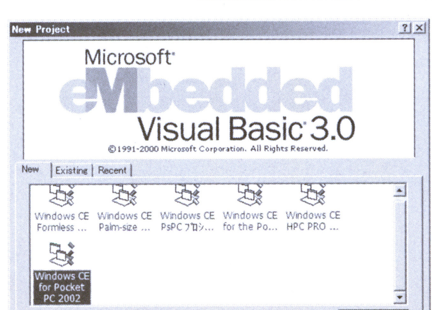
eVBを使って，シンプルなスクロールゲームを作ってみました。キャラクターを赤い地面に落とさず，どこまで持ちこたえられるか，という内容です（**写真9，写真10**）。タッチパネル上でスタイラスをドラッグさせると，その位置へキャラクターは移動しようとします。またスタイラスを離すとジャンプします。



《写真9》サンプル・ゲームの動作画面。赤い地面に落ちないようスタイラスで誘導する



《写真10》スタイラスを離すとジャンプ。赤い地面をうまく飛び越えよう



《写真11》ProjectはPocketPCもしくはPocketPC 2002を選択する

ゲームが進むにつれてスクロール速度は上がっていき，難しくなっていきます。

それではさっそくプログラミングをしてみましょう。まずeVBを起動し，New Projectから“Windows CE for Pocket PC”もしくは“Windows CE for Pocket PC2002”を選択します（**写真11**）。フォームにタイマーコントロールを配置し，そのIntervalの項目で50程度の値を設定。フォームのScaleModeプロパティを“3-Pixel”にして，リストを入力していきます。

プログラムの内容は普通のVBに近いもので，APIのBitBltを利用してキャラクターなどを表示しています。ただeVBでは，ピクチャー・ボックスがBitBlt用に使うには機能がちょっと貧弱だったので，代わりにAPIのCreateBitmapを利用してビットマップを作成しています。このCreateBitmapを使ったせいか，エミュレータだと表示がうまくいかないようです。そのときはPocketPC実機で試してみてください。

PocketPCでは，BitBltのようなGDIを利用した画面表示以外に，“GAPI”と呼ばれるDirectXのように直接画面を制御できる仕組みがあります。高速スクロールをするゲームを作る場合などには有用でしょう。

なお，API使用で呼び出すDLLファイルが異なるため，このリストそのままでは，VB6.0などで動作はできません。ただ少ない変更でVB6.0用にできますので，腕に覚えのある人は解析してみてもいいでしょう。

まとめ

発売当初は安価なPalm製品と多機能なPocketPCというような区分けがされていましたが，豊富な本体ラインアップや機能追加により，両者の垣根はなくなりつつあります。デスクトップパソコンとの連携が不可欠な点も同じ。ベーマガ読者なら，これまでVBやVCで培ったノウハウを活かせるぶん，PocketPCの方が扱いやすいかもしれません。

```
Option Explicit
'APIの宣言
Declare Function GetDC Lib "Coredll" (ByVal hwnd As
Long) As Long
Declare Function ReleaseDC Lib "Coredll" (ByVal hwnd
As Long, ByVal hDC As Long) As Long
Declare Function CreateBitmap Lib "Coredll" (ByVal
nWidth As Long, ByVal nHeight As Long, ByVal nPlanes
As Long, ByVal nBitCount As Long, lpBits As Byte) As
Long
Declare Function CreateCompatibleDC Lib "Coredll"
(ByVal hDC As Long) As Long
Declare Function DeleteObject Lib "Coredll" (ByVal
hObject As Long) As Long
Declare Function DeleteDC Lib "Coredll" (ByVal hDC As
Long) As Long
Declare Function SelectObject Lib "Coredll" (ByVal hDC
As Long, ByVal hgdiobj As Long) As Long
Declare Function BitBlt Lib "Coredll" (ByVal hDestDC
As Long, ByVal X As Long, ByVal Y As Long, ByVal
nWidth As Long, ByVal nHeight As Long, ByVal hSrcDC As
Long, ByVal xSrc As Long, ByVal ySrc As Long, ByVal
dwRop As Long) As Long
Const srccopy = &HCC0020
Const srcinvert = &H660046
Const srcand = &H8800C6

'オフスクリーンバッファ用の変数
Dim hChrDC As Long, hChrBitmap As Long, hBitmap1 As
Long, bitmap1(30719) As Byte
Dim hOffDC As Long, hOffBitmap As Long, hBitmap2 As
Long, bitmap2(230399) As Byte
Dim hBackDC As Long, hBackBitmap As Long, hBitmap3 As
Long, bitmap3(230399) As Byte
Dim Brush As Long, ClrRect(3) As Long
'API呼び出しよう
Dim ret As Long
'ゲーム中の座標など
Dim mx As Integer, my As Integer
Dim mox As Integer, moy As Integer
Dim jy As Integer, jf As Boolean, jt As Integer
Dim fild(7, 10) As Integer
Dim scrp As Integer, scrsp As Integer
Dim rndp As Integer
Dim sc As Long
Dim Keys(255) As Boolean
Dim firstPaint As Boolean


Private Sub Form_Load()
Dim hDesktopDC As Long
'起動時初期化
  firstPaint = True
  'オフスクリーン用のデバイスコンテキストを作成する
  hDesktopDC = GetDC(0)
  hChrDC = CreateCompatibleDC(hDesktopDC)
  hChrBitmap = CreateBitmap(160, 64, 1, 24,
bitmap1(0))
  hBitmap1 = SelectObject(hChrDC, hChrBitmap)
  hOffDC = CreateCompatibleDC(hDesktopDC)
  hOffBitmap = CreateBitmap(240, 320, 1, 24,
bitmap2(0))
  hBitmap2 = SelectObject(hOffDC, hOffBitmap)
  hBackDC = CreateCompatibleDC(hDesktopDC)
  hBackBitmap = CreateBitmap(240, 320, 1, 24,
bitmap3(0))
  hBitmap3 = SelectObject(hBackDC, hBackBitmap)
  ret = ReleaseDC(0, hDesktopDC)
  '座標初期化
  mx = 160
  my = 200
  mox = mx
  moy = my
  scrp = 0
  scrsp = 1
  sc = 0
  rndp = 30
End Sub

Private Sub Form_MouseDown(Button As Integer, Shift As
Integer, X As Single, Y As Single)
  mox = X
  moy = Y
End Sub

Private Sub Form_MouseMove(Button As Integer, Shift As
Integer, X As Single, Y As Single)
  mox = X
  moy = Y
End Sub

Private Sub Form_MouseUp(Button As Integer, Shift As
Integer, X As Single, Y As Single)
  If jf = False Then
    jf = True
    jt = 8
  End If
End Sub

Private Sub Form_OKClick()
'終了
  'DC関係の解放
  ret = DeleteObject(Brush)
  hChrBitmap = SelectObject(hChrDC, hBitmap1)
  DeleteObject (hChrBitmap)
  DeleteDC (hChrDC)
  hOffBitmap = SelectObject(hOffDC, hBitmap2)
  DeleteObject (hOffBitmap)
  DeleteDC (hOffDC)
  hBackBitmap = SelectObject(hBackDC, hBitmap3)
  DeleteObject (hBackBitmap)
  DeleteDC (hBackDC)
  App.End
End Sub

Private Sub Form_Paint()
Dim ret As Long, i As Integer, j As Integer, k As
Integer
Dim dat As String, enc As String, col(7) As Long
Dim a As Integer, c As Integer
  'キャラクター作成
  If firstPaint = True Then
    firstPaint = False
      dat = "Z0Z0F0Z1C1Z2Z2E2Z0Z0F03Z444A1Z2Z2"
dat = dat + "D2Z0Z0F03Z444B1Z2Z2C2K0C5Z0P03Z44"
dat = dat + "4C1Z2Z2B2K0C5Z0P03Z444D1Z2Z2A2K0C"
dat = dat + "5Z0P03Z444E1Z2Z222K0C5Z0P03Z444F1"
dat = dat + "Z2Z22M05Z0R03Z444G1Z2Z2M05Z0"R03Z"
dat = dat + "444H1Z2Y2J0E5Z0O03Z444I1Z2X2M05Z0"
dat = dat + "R03Z444J1Z2W2M05Z0R03Z444K1Z2V2M0"
dat = dat + "5Z0R03Z444L1Z2U2L0505Z0Q03Z444M1Z"
dat = dat + "2T2K05A05Z0P03Z444N1Z2S2J05C05Z0O"
dat = dat + "03Z444O1Z2R2Z0Z0F03Z444O1Z2R2Z0Z0"
dat = dat + "F03Z444O1Z2R2Z0Z0F03Z444O1Z2R2Z0Z"
dat = dat + "0F03Z444O1Z2R2Z0Z0F03Z444O1Z2R2Z0"
dat = dat + "Z0F03Z444O1Z2R2Z0Z0F03Z444O1Z2R2Z"
dat = dat + "0Z0F03Z444O1Z2R2Z0Z0F03Z444O1Z2R2"
dat = dat + "Z0Z0F03Z444O1Z2R2Z0Z0F03Z444O1Z2R"
dat = dat + "2Z0Z0F03Z444O1Z2R2Z0Z0F03Z444O1Z2"
dat = dat + "R2Z0Z0F03Z444O1Z2R2Z0Z0F03Z444O1Z"
dat = dat + "2R2Z0Z0F0Z3A3O1Z2R2Z1Z1F1Z6B6N1Z2"
dat = dat + "R2Z1Z1F10Z6B6M1Z2R2Z1Z1F10OZ6B6L1"
dat = dat + "Z2R2K1C0Z1P1A0Z6B6K1Z2R2K1C0Z1P1B"
dat = dat + "0Z6B6J1Z2R2K1C0Z1P1C0Z6B6I1Z2R2K1"
dat = dat + "C0Z1P1D0Z6B6H1Z2R2M10Z1R1E0Z6B6G1"
dat = dat + "Z2R2M10Z1R1F0Z6B6F1Z2R2J1E0Z1O1G0"
dat = dat + "Z6B6E1Z2R2M10Y1F0K1H0Z6B6D1Z2R2M1"
dat = dat + "0V1L0H1I0Z6B6C1Z2R2M10T1P0F1J0Z6B"
dat = dat + "6B1Z2R2L1010S1P0F1K0Z6B6A1Z2R2K10"
dat = dat + "A10Q1R0E1L0Z6B611Z2R2J10C10P1R0E1"
dat = dat + "M0Z6B61Z2R2Z1H1P0F1N0N1Z2Z2F2Z1H1"
dat = dat + "P0G1N0M1Z2Z2F2Z1J1L0J1N0L1Z2Z2F2Z"
dat = dat + "1M1F0N1N0K1Z2Z2F2Z1Z1J1N0J1Z2Z2F2"
dat = dat + "Z1Z1K1N0I1Z2Z2F2Z1Z1L1N0H1Z2Z2F2Z"
dat = dat + "1Z1M1N0G1Z2Z2F2Z1Z1N1N0F1Z2Z2F2Z1"
dat = dat + "Z1O1N0E1Z2Z2F2Z1Z1P1N0D1Z2Z2F2Z1Z"
dat = dat + "1Q1N0C1Z2Z2F2Z1Z1R1N0B1Z2Z2F2Z1Z1"
dat = dat + "S1N0A1Z2Z2F2Z1Z1T1N011Z2Z2F2Z1Z1U"
dat = dat + "1N01Z2Z2J2"

    col(0) = 0
    col(1) = &HFFFFFF
    col(2) = 255
    col(3) = 8421504
    col(4) = 12632256
    col(5) = 65535
    col(6) = 4210752
    enc = "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    'データを展開してフォームに一旦描く
    i = 0: j = 0
    While Len(dat) > 0
      a = InStr(enc, Mid(dat, 1, 1)) - 1
      If a > 7 Then
        c = col(InStr(enc, Mid(dat, 2, 1)) - 1)
        dat = Mid(dat, 3)
        For k = 1 To a - 7
          PointSet i, j, c
          i = i + 1
        Next k
      Else
        dat = Mid(dat, 2)
        c = col(a)
        PointSet i, j, c
        i = i + 1
      End If
      If i > 159 Then i = i - 160: j = j + 1
    Wend
    'キャラクター用BITMAPに転送
    ret = BitBlt(hChrDC, 0, 0, 160, 64, Form1.hDC, 0,
0, srccopy)
    'フィールドを初期化
    For j = 0 To 10
      For i = 0 To 7
        fild(i, j) = 1
        ret = BitBlt(hBackDC, i * 32, j * 32, 32, 32,
hChrDC, 64, 0, srccopy)
      Next i
    Next j
  End If
End Sub


Private Sub Timer1_Timer()
Dim i As Integer, j As Integer, k As Integer, l As
Integer
'タイマー処理
  '自機移動
  If Abs(mx - mox) < 8 Then
    mx = mox
  Else
    mx = mx + Sgn(mox - mx) * 8
  End If
  If Abs(my - moy) < 8 Then
    my = moy
  Else
    my = my + Sgn(moy - my) * 8
  End If
  'ゲームオーバー判定
  If fild((mx - 15) / 32, ((my - 15 + 320 - scrp) Mod
320) / 32) = 0 And jf = False Then
    MsgBox sc
    Timer1.Enabled = False
    Call Form_OKClick
  End If
  'ジャンプ
  If jf = True Then
    If jt > 0 Then
      jy = jy - 3
      jt = jt - 1
    Else
      jy = jy + 3
      If jy >= 0 Then
        jf = False
      End If
    End If
  End If
  'スクロール処理
  sc = sc + 1
  If (sc Mod 250) = 0 Then
    scrsp = sc / 250
    If scrsp > 16 Then scrsp = 16
    rndp = rndp - 1
    If rndp = 0 Then rndp = 1
  End If
  scrp = scrp + scrsp
  If scrp >= 320 Then scrp = scrp - 320
  If (scrp Mod 32) = 0 Then
    j = (9 - scrp / 32) Mod 10
    For i = 0 To 7
      ret = BitBlt(hBackDC, i * 32, j * 32, 32, 32,
hChrDC, 128, 32, srccopy)
    Next i
    For i = 0 To 7
      k = Int(Rnd(1) * (rndp / 10))
      fild(i, j) = k
      If k > 0 Then
        ret = BitBlt(hBackDC, i * 32, j * 32, 32, 32,
hChrDC, 64, 0, srccopy)
      End If
    Next i
  End If
  'オフスクリーンを消去
  ret = BitBlt(hOffDC, 0, scrp, 240, 320 - scrp,
hBackDC, 0, 0, srccopy)
  ret = BitBlt(hOffDC, 0, 0, 240, scrp, hBackDC, 0,
320 - scrp, srccopy)
  '各キャラクターを転送
  ret = BitBlt(hOffDC, mx - 15, my - 15, 32, 32,
hChrDC, 32, 32, srcand)
  ret = BitBlt(hOffDC, mx - 15, my - 15, 32, 32,
hChrDC, 32, 0, srcinvert)
  ret = BitBlt(hOffDC, mx - 15, my - 15 + jy, 32, 32,
hChrDC, 0, 32, srcand)
  ret = BitBlt(hOffDC, mx - 15, my - 15 + jy, 32, 32,
hChrDC, 0, 0, srcinvert)
  'オフスクリーンから画面に転送
  ret = BitBlt(Form1.hDC, 0, 0, 240, 320, hOffDC, 0,
0, srccopy)
End Sub
```

PocketPC 新製品情報

画面が見やすく長時間運用可能！！



カシオペア E-3000

オープン・プライス（実売価格60,000円前後）

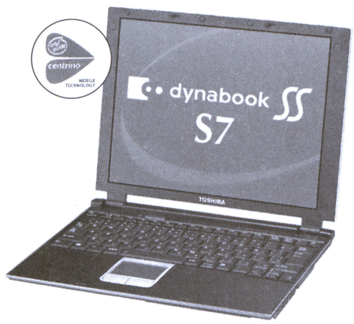
メーカー	カシオ計算機
URL	http://www.casio.co.jp/
OS	Pocket PC 2002 Software日本語版
CPU	IntelR PXA255 400MHz
メモリ	SDRAM64Mバイト
ディスプレイ	240×320ドット半透過型TFTカラー液晶
主なインターフェース	CFカード（タイプⅡ）スロット×1/SDカード・MMCスロット×1/USB/赤外線通信
問い合わせ先	お客様相談室☎03-5334-4611

2種類のメモリカードスロットを備えるほか，屋内外どちらでも見やすい半透過液晶や最新CPU PXA255採用による低消費電力を実現。バッテリ交換による長時間運用を可能にするなど，使いやすさを主眼にパワーアップが図られた最新モデルです。

インフォ・キャッチ Information Catch Up 05

新モバイル・テクノロジ“セントリーノ”搭載マシン，続々登場!!

ノートパソコン
ワイヤレスLANを使いやすくしたノートパソコン
東芝　dynabook SS S7



(S7-290LNKW)
●4月中旬発売
オープン価格（実売予想価格：22万円前後）
CPU：PentiumM 900MHz
メモリ：256MB
ハードディスク：40GB
ディスプレイ：12.1インチ TFT液晶
ドライブ：オプション
バッテリ駆動時間：約2.5時間
■問い合わせ先
ナビダイヤル　0570-00-3100
http://dynabook.com/

ノートパソコン
大切なデータを安全に扱うためのセキュリティチップ搭載
IBM　ThinkPad X31



(2672-JHJ)
IBMダイレクト価格：25万4,000円
CPU：PentiumM 1.4GHz
メモリ：256MB
ハードディスク：40GB
ディスプレイ：12.1インチ TFT液晶
ドライブ：オプション
バッテリ駆動時間：約5.2～6時間
■問い合わせ先
☎0120-04-1992
http://www.ibm.com/jp/pc/

ノートパソコン
11時間を越えるバッテリ駆動時間を実現
NEC　Lavie M LM500/5D

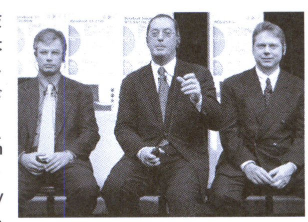


オープン価格（実売価格：25万円前後）
CPU：PentiumM 1.3GHz
メモリ：256MB
ハードディスク：約40GB
ディスプレイ：12.1インチ TFT液晶
ドライブ：CD-R/RW,DVD-ROM
バッテリ駆動時間：約4.7時間
■問い合わせ先
☎0120-977-121
http://121ware.com/lavie/m/

インテルCentrino（セントリーノ）モバイル・テクノロジ
高性能なのに省電力。ワイヤレス標準時代のノートパソコン用テクノロジが誕生。

3月12日，インテルのモバイルPC用の新技術“Centrino”が世界で同時発表され，日本でもNECや東芝，IBMなど各パソコン・メーカーからセントリーノ搭載のノートパソコンが同時発表されました。

Centrinoとは，モバイル用CPU“Pentium（ペンティアム） M”と，その性能を最大限に活かす“855チップセット”そして無線LANモジュール“PRO/Wireless（ワイヤレス）2100”の三つがセットになったもので，従来のモバイルPentium4よりも低い動作クロック数での高速処理を実現し，さらにバッテリーの持続時間も向上させています。



▲報道陣の質問に答えるインテル コーポレーションのポール・S・オッテリー二社長（中央）とダディ・パルムッター副社長（左），そしてインテル株式会社ジョン・アントン社長（右）

タブレットPC
スタイラスペンで操作できるタブレットPC
ビューソニック　Tablet PC V1100



オープン価格（実売価格：25万円前後）
CPU：モバイルPentiumⅢ 866MHz
メモリ：256MB
ハードディスク：20GB
ディスプレイ：10.4インチ TFT液晶
※デスクトップパソコンとして使用する場合，別売ドッキングステーション（実売予想価格2万円前後）が必要となります。
■問い合わせ先
電子メール service@jp.viewsonic.com
http://www.viewsonicjapan.co.jp/

CD-RWドライブ
上ぶた開閉式のスタイリッシュなCD-RWドライブ
アイオメガ　Predator 40×24×40



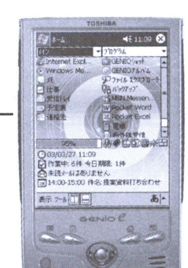
価格：1万4,900円
対応OS：Windows XP/Me/98SE/2000，MacOS X
対応メディア：CD-R/RW
ドライブスピード：記録40倍速，書込24倍速，読出40倍速
■問い合わせ先
☎044-752-6651
http://www.iomega.com/jp/

オーディオレコーダ
厚さ9.5mmの薄型オーディオレコーダ
ソニックブルー　Rio DR30



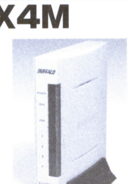
オープン価格（実売価格：2万4,800円）
最大録音時間：約36時間
再生：MP3,WMA　録音：MP3
■問い合わせ先
ナビダイヤル 0570-00-2111
http://www.sonicblue.co.jp/

PDA
ポケットPC初のデジカメ内蔵モデル
東芝　GENIO e550C



オープン価格（実売価格：7万円前後）
OS：Pocket PC 2002
ディスプレイ：4インチ TFT液晶
有効画素数：約31万画素
カメラフラッシュ：内蔵
■問い合わせ先
☎043-279-2631
http://genio-e.com/

ブロードバンドルータ
不正アクセス検出機能を備えたBBルータ
メルコ　BLR-TX4M



価格：6,980円
転送速度：最大65Mbps
サポート規格：IEEE802.3/802.3u準拠
■問い合わせ先
☎03-5326-3754
http://www.melcoinc.co.jp/

モバイルプリンタ
手のひらサイズのモバイルプリンタ
ブラザー工業　MW-100e



(Palm OS対応)
●5月初旬発売予定
オープン価格（実売予想価格：3万円前後）
解像度：300dpi
本体サイズ：幅100×奥行160×高17.5mm
■問い合わせ先
☎052-824-3091
http://www.brother.co.jp/

電子辞書

キヤノン　ワードタンクシリーズ　<<<　英語を学習中の人なら



IDF-2200E

価格：2万4,800円
辞書数：大修館書店「ジーニアス英単語2500改訂版」他4冊
サイズ：幅117×奥行72×厚17.5mm
重さ：約140g

大学受験に最適な辞典を収録
英英・英和・和英，英単語テスト機能つき



IDF-4600

価格：4万5,000円
辞書数：大修館書店「新実用英語ハンドブック」他8冊
サイズ：幅140×奥行94×厚14mm
重さ：約205g

ビジネス英語に最適な辞典を収録
正しい英語で仕事の即戦力に

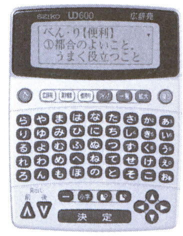
セイコー　フルコンテンツ電子辞書　<<<　使いやすさで選びぶなら



SR-T4020

価格：3万2,000円
辞書数：三省堂「全訳読解古語辞典」他5冊
サイズ：幅137×奥行88×厚15.4mm
重さ：約210g

古語辞典・漢和辞典をコンパクトに収録
持ち運びに便利で高校生向けの電子辞書



UD600

価格：2万2,000円
辞書数：岩波書店「広辞苑」1冊
サイズ：幅147×奥行190×厚38mm
重さ：約350g

広辞苑をまるまる収録
50音配列の大きなキーで簡単入力



CL500

価格：1万2,000円
辞書数：オリジナル「漢字辞典」1冊
サイズ：幅110×奥行145×厚34.5mm
重さ：約190g

大きな文字表示が魅力
50音キーと数字キーを独立して配置

カシオ　エクスワードシリーズ　<<<　持ち運びに便利な携帯性で選ぶなら



XD-R7100

価格：4万5,000円
辞書数：クラウン「独和辞典」他8冊

英語以外の外国語辞典を収録
独語・仏語・中国語に対応

サイズ：幅139×奥行96.5×厚12.7mm
重さ：約215g



XD-R970

価格：3万6,000円
辞書数：岩波書店「広辞苑」他21冊

ビジネス・学習・趣味・旅行に
小型モデルで22種の辞書を収録

サイズ：幅115×奥行79×厚12.4mm
重さ：約130g



XD-V6200

価格：4万3,000円
辞書数：アイ・ティ・コム「超訳IT用語辞典」他22冊

スタイリッシュタイプ
薄さ9.8mmの着せ替え自由

サイズ：幅140.5×奥行98×厚9.8mm
重さ：約200g

■問い合わせ先（キヤノン販売）
ナビダイヤル 0570-01-9000
http://cweb.canon.jp/wordtank/

■問い合わせ先（セイコーインスツルメンツ）
☎0120-052-440
http://www.sii.co.jp/cp/index.html

■問い合わせ先（カシオ計算機）
ナビダイヤル 0570-088901
http://www.casio.co.jp/exword/

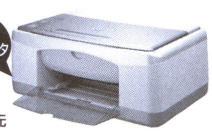
複合機　スキャナ・コピー・プリンタ機能を一体化

▼NEC　ピクティ500 PR-M500
オープン価格（実売価格:2万円前後）



■問い合わせ先
☎0120-977-121
http://www.nec.co.jp/

▼日本HP　hp psc 1210
オープン価格（実売価格:2万円前後）



■問い合わせ先
☎03-5304-6777
http://www.hp.com/jp/

インフォシーク有料ホームページサービスを開始

総合ポータルサイト「インフォシーク」では，従来の無料ホームページサービス「infoseek isweb」の有料版「infoseek isweb ベーシック」を開始しました。有料バージョンでは，ページ内の広告がすべて表示されなくなり，容量が50MBから300MBに増量しています。

↓詳しい情報はこちらまで

http://isweb.infoseek.co.jp/

ソフトいろいろ

市川ソフトラボラトリー
肌のシミやしわを補正してくれる
"メーキャップフィルタ"を搭載
デイジーデジカメビューティ
価格：6,400円



■問い合わせ先
☎043-296-8075
http://www.isl.co.jp/

エー・アイ・ソフト
デジカメde!!フォト工房2
価格：1万1,800円



■問い合わせ先
☎03-3376-7122
http://www.aisofy.co.jp/

ジャングル
ホームページ作成王2004
価格：6,800円



■問い合わせ先
☎03-3234-2102
http://www.junglejapan.com/

ユーリード
Ulead VideoStudio 7
価格：1万4,800円



■問い合わせ先
☎03-5491-5661
http://www.ulead.co.jp/

ソースネクスト
特打ファンタジー
くまのプーさん
価格：1,980円



■問い合わせ先
☎03-5350-4844
http://www.sourcenext.com/

TDK
HDデータクリア2
＋メモリカードクリア
価格：7,800円



■問い合わせ先
☎03-5201-7156
http://www.tdk.co.jp/

記録型パソコン用DVD技術を知る！



第4回

オーサリング&ライティング

遠藤　正奥

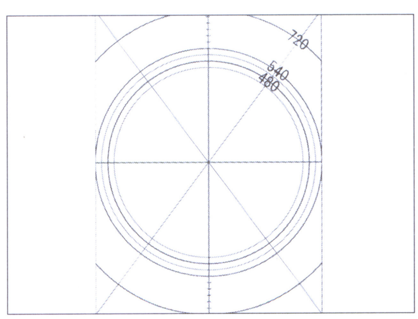
前回で，編集が完了したビデオ素材が用意できましたので，オーサリングとライティングソフトを使って，DVDビデオを完成させます。プロユースのカノープスの編集システムの紹介は割愛させていただきます。

記録型DVDディスクにビデオ信号を，データとして書き込むのは難しいことではありません。しかし，パソコンのデータとしての記録ではパソコンでしか見られず，使える範囲も限られてしまいます。一般家庭に普及しているDVDプレーヤで再生できるDVDビデオができれば，使い道は広がります。DVDビデオの規格に準拠した再生互換性の高いDVDビデオで，しかもメニューが使いやすく，ディスクメディアの特長である，見たいところが直ぐ見られるディスクに仕上げるためには，編集の後のオーサリングが重要な役割を果たします。ここでは，簡単な操作で誰でも間違いなくDVDビデオができるソニックソルーションズ社の「MyDVD」と，本格オーサリングソフト「DVDit」の2本のソフトを使って，それぞれでDVDビデオを完成させていきます。

静止画の画像サイズ

DVDビデオの最高解像度(フレームサイズ)は720ピクセル×480ピクセル(NTSC規格)になります。DVカメラからキャプチャしたビデオファイルは，この解像度になります。標準のテレビ画面のアスペクト比は4:3ですから変だなと思われるかもしれませんが，NTSC規格のテレビのピクセルは縦に長くなっていますので，720ピクセル×480ピクセルで，表示画面は4:3になります。一方，パソコンのモニターのピクセルは正方形ですので，720ピクセル×540ピクセルで4:3になります。

今回，編集に使用しましたShowBizのプレビューウインドは，テレビ画面と同じアスペクト比になるように補正されていませんので，DVカメラからキャプチャした画像はモニター画面上では横長になります。もちろん，DVDに書き出した画像はテレビ画面で正常なアスペクト比になります。



《画面１》縦位置の画像は，このように解像度とカンバスサイズを変更して720×540（ピクセル）の画像に加工

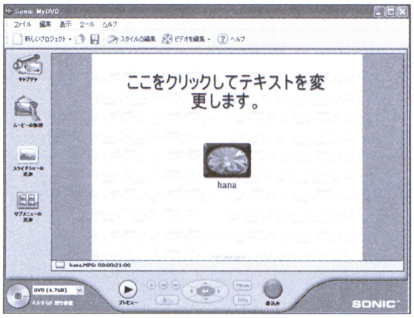
デジタルカメラ，スキャナから取り込んだ画像や，パソコンで作ったグラフィックのイメージをShowBizのムービーの中に挿入するときは，画像の解像度(ピクセル数)を画像処理ソフトによって変更する必要があります。

これからオーサリング·書き込みに使う「MyDVD」「DVDit」は，静止画をムービーとムービーの間に挿入したり，メニュー画面に使ったりできます。これらのソフトのプレビュー画面はテレビ画面と同じ4:3のアスペクト比で表示されますので，720ピクセル×540ピクセルとします。静止画像の扱いはソフトにより違うことがありますので，アスペクト比が正確で最高画質が得られるように画像処理ソフトで加工します。「DVDit」はすべての画像を4:3に変換しますので，縦位置の画像をそのまま使うと横方向に大きく伸張された画面になってしまいます(画面1)。

MyDVDを使って DVDビデオを作る

ムービーを取り込む

「ShowBiz」で編集後にメニューの「作成」から「MyDVDプロジェクトに書き出し」を選択し，ファイル名，保存場所



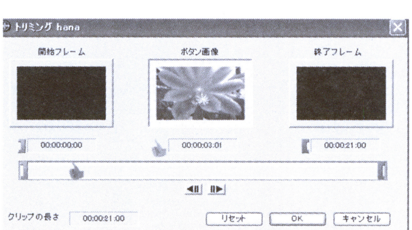
《画面2》ボタンフレームに取り込まれたムービーのサムネイルが表示される

を指定すると，ムービーが保存されると同時にMyDVDが立ち上がり，ムービーが取り込まれた状態になります(**画面2**)。

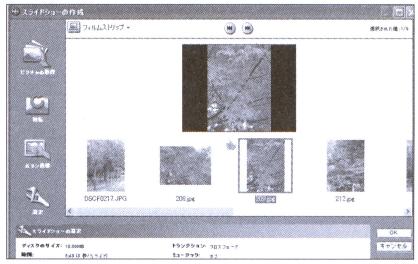
また，「ムービーの作成」でDVDファイル形式で保存した場合は，MyDVDを立ち上げ，「DVD Videoプロジェクトの作成または修正」を選び，次の画面で「ムービーの取得」から保存したファイルを選ぶと同じ画面になります。

これで一つのムービーが取り込まれました。表示されるサムネイルは，ムービーの開始に近いところから選ばれていますので，フェードインなどのトランジションが入れてある場合は，その効果が出た画像になってしまいます。サムネイルをダブルクリックするとトリミング・ウインドウが現れますので，サムネイルスライダーを動かして表示画面を選びます(**画面3**)。

取り込んだムービーやクリップの再編集は，ツールバーの「ビデオを編集」をクリックすると「ShowBiz」が立ち上がり，選択したクリップまたは表示されているメニューのすべてのクリップが送られます。「ShowBiz」がインストールされている場合のみ，このボタンは表示されます。テキストの挿入，他のムービーやクリップとの色調合わせなどに使かえる便利な連携機能です。



《画面3》表示するサムネイルは好きなところが選べる



《画面4》静止画を次々に表示できるスライドショーが作れる

「スライドショー」を作る

デジタルカメラやスキャナで取り込んだ画像ファイルを次々に表示する「スライドショー」を作ることができます(**画面4**)。メインウィンドウで「スライドショーの追加」をクリック，「スライドショウの作成」，ウインドウで「ピクチャの取得」をクリックして画像を選びます。画像の削除，回転とサムネイルに使う画像の選択は画像を右クリックして行います。さらに「設定」で，表示時間，オーディオトラック，トランジション，レターボックスの色などが設定できます。

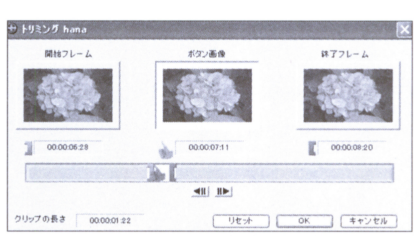
ここでは元の静止画の解像度に関係なく縦横比はそのまま維持され，縦か横が画面いっぱいになるように調整されます。

メニューを作る

ディスクメディアの大きな特長の一つであるダイレクトアクセスを生かして，途中の好きなところからの再生ができるようにメニュー画面を作ります。これに対応するには二とおりの方法があります。

①取り込んだムービーをトリミングする

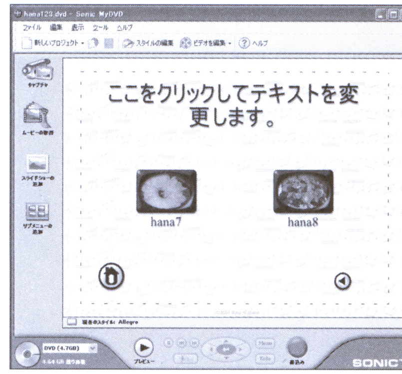
MyDVDは，取り込んだムービーのトリミングができます。ボタンをダブルクリックするとトリミング画面が現れます。ここで，開始位置と終了位置，それにサムネイルとして表示する部分を決めることができます(**画面5**)。再度同じムービーを取り込み，これを繰り返すことにより複数のクリップに分けることができます。しかし，この方法では画面切り替えのトランジションが使えません。また作業としては，かなり手間のかかることになります。



《画面5》トリミングでクリップを抜き取ることができる



《画面6》ボタンが6個以上になると次のメニュー画面が作られる

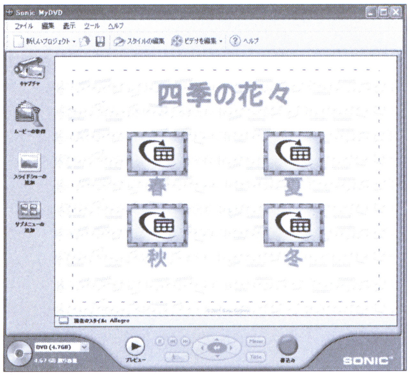


《画面7》2ページ目のメニュー

② 編集段階で一区切りごとのクリップで保存する

これは，ちょうど市販DVDビデオのチャプターに相当する範囲を独立したファイルとして，「ムービーの作成」から保存しておきます。クリップの始まりと終わりは次のクリップとのつながりを考えて，映像・音声ともにフェードイン，フェードアウトなどのトランジションで，つながりが不自然にならないように考えておきます。一つのムービーというよりはクリップごとの保存になります。メニュー画面には六つのボタンまで表示され，クリップの数がそれ以上になると次のメニュー画面が作られます(**画面6,7**)。

また，サブメニューも作れます。メニューに階層をつけることで，ビデオクリップをわかりやすく整理することができます。年間の記録なら行事ごとにサブメニューを作り，そこにクリップを配置したメニューを作るなどします。例では季



《画面8》テーマごとにサブメニューを使うと，さらにサーチしやすくなる

節ごとに分けてサブメニューを作り，そこに花ごとのクリップを配置した，花のアルバムを作ってみました(**画面8**)。

メニューの編集

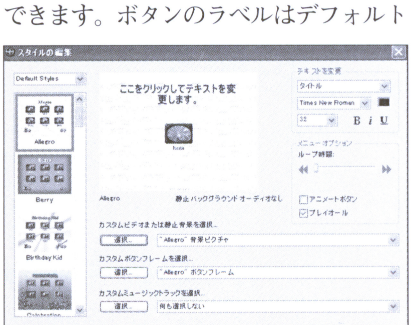
ツールバーから「スタイルの編集」ウィンドウを開きます(**画面9**)。

①「メニュー画面の背景」が27種類用意されていますが，好みの静止画像やビデオクリップを使用することもできます。ビデオクリップは，先頭の1秒から60秒までが再生可能で，メニューオプションのループ時間のスライダーで調整します。ここで使う画像の解像度は前述のように静止画は720×540(ピクセル)です。

②「ボタンフレーム」は15種類用意されています。

③「ミュージックトラック」を設定できます。メニュー画面表示中にサウンドを再生することができるもので，メニューの背景が静止画のときはオーディオクリップ全体が繰り返し再生され，動画のときは動画と同じ時間だけ,最初の部分が再生されます。

④ メニューで使用するタイトルとボタンの文字の種類，大きさ，カラーを設定できます。ボタンのラベルはデフォルトではファイル名が表示されますので，メニューにふさわしい表示にします。



《画面9》メニュースタイルの編集ウインドウ

⑤「アニメートサムネイル」を有効にすると，ボタンにリンクしたクリップから抜粋した画像が動きます。このクリップは最大60秒間設定できますが，背景に動画を使っている場合は同じ時間になります。

⑥「プレイオール」をオンにすると，一つのクリップの再生が終わると次のクリップを再生し，次々とすべてのビデオクリップを順番に再生し，最後まで再生してメニューに戻ります。この機能をオフにすると，選んだクリップを再生した後メニューに戻るようになります。

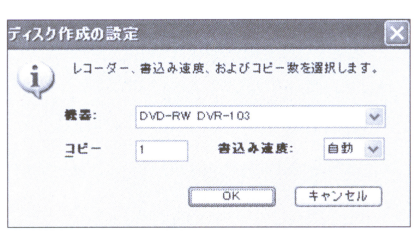
作品をチェック

ディスクへ書き込む前に，実際にどのように見えるのかプレビューします。プレビューコントロールボタンは，DVDプレーヤのリモコンにある主要ボタンに用意されています。メニューにビデオ背景や，ボタンにビデオサムネイルを使っている場合は，プレビュー用のレンダリングデータを 作成する必要があります。「レンダリング」ボタンをクリックすることで，動画メニューデータのレンダリングが行われます。レンダリングを行わないと表示は静止画になります。

メニューからのサーチなどの機能，画像と音声をよくチェックします。不具合があれば納得のいくまで再編集するなどして完成度を高めます。

ディスクへの書き込み

いよいよディスクに書き込みます。まずはMyDVDでの設定や使用ファイルなどの情報をプロジェクトファイルに保存します。このときプロジェクトファイルに付けた名前が，DVDディスクのボリューム名になります。



《画面10》OKをクリック，書き込みが終了しイジェクトされるのを待ちます

赤い「書き込み」ボタンをクリック，「ディスク作成の設定」はデフォルトでよいはずです。OKボタンをクリックすると書き込みが始まります(**画面10**)。高速書き込みのCD-Rになれていると，等速で1時間はとても長く感じられます。最新の4倍速メディアとドライブを使えば，かなり快適になります。

最初は，書き換えができるリライタブルディスク(DVD-RW)を使います。パソコンの設定の問題で書き込みに失敗するかもしれませんし，DVDプレーヤでの互換性のチェックもしたほうがよいでしょう。最近のDVDプレーヤはDVD-RWに対応しているはずですが，古いモデルでは再生できないものもありますので確認してください。DVD-Rのほうが互換性は高いのですが，DVD-RWで試してからにします。

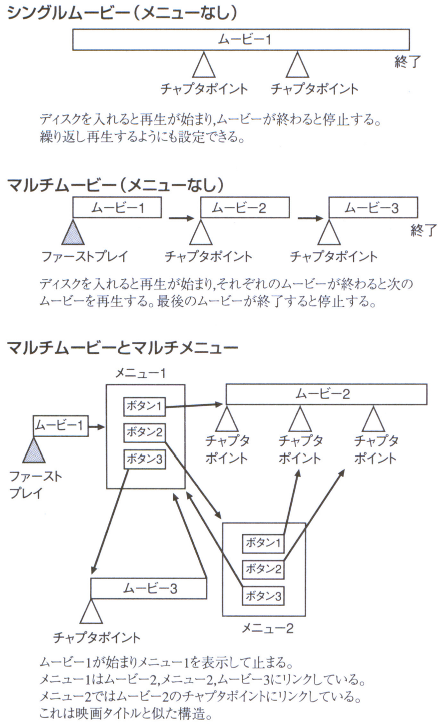
「MyDVD」には，今回作業してきました「DVD Videoプロジェクトの作成または修正」の他に，ダビング感覚でDVカメラからDVDが作れる「Direct-to-DVDにビデオを転送」と，リライタビルディスクに書き込んだDVDの内容を変更できる「既に作成済みのOpenDVDディスクを編集」という機能をもっています。

簡単な操作で階層メニュー構造が作れるなど，エントリーユーザーには使いやすいオーサリングソフトです。アークソフトの「ShowBiz」との連携で，かなりのレベルの作品を作ることができます。徹底的に使い込んでみていただきたいと思います。そして，さらに高度な編集テクニックを駆使した作品を作りたいのであれば，プロユースの編集ソフトに挑戦してください。

DVDitを使ったオーサリング

MyDVDとDVDitの違い

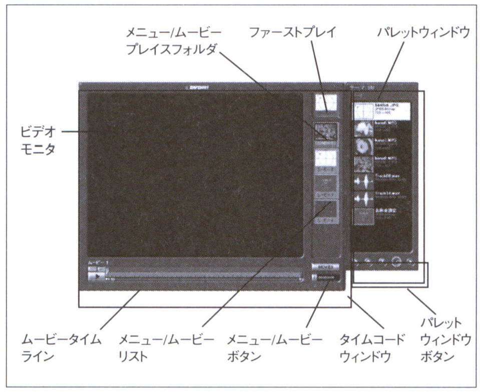
「MyDVD」には，キャプチャ機能，簡単なビデオ編集(トリミング)機能，編集ソフトとの連携機能などをもっています。また，ほとんどのオーサリング機能が自動的に処理されますので，より簡単に



《第1図》メニュー構造の例

《第1表》推奨される画質を考えたビデオ，画像サイズ

ファイル形式	画像サイズ(ピクセル)
MPEG-2	720×480
AVIおよびQuickTimeビデオ	矩形のピクセルのビデオ用　720×480 正方形のピクセルのビデオ用　720×540
メニューの背景の画像	640×480
静止画像	720×540
ボタンサイズ	72×60（最小）＊拡大は劣化大 300×80（推奨）



《画面11》DVDitのメインワークスペース，多機能だが操作性に優れている



《画面12》パレットウインドウに素材をインポート

DVDを作りたいという方にピッタリのソフトです。

一方,「DVDit」はオーサリング専用ソフトです。ほとんどの機能を手動で設定しなければなりませんが，その分，各メニューを異なるデザインにしたり，画面の任意の場所にテキストやボタンリンクなどのメニューオブジェクトを配置することができます。また，ナビゲーションコマンドを手動で指定することで，ナビゲーションを作る側が自由にコントロールできます。作る側の意図を生かした，より高度なDVDオーサリングが行えます。

多彩なメニュー構造が可能

素材の内容，使用目的などに応じて，さまざまなナビゲーション動作ができるメニュー構造が作れます(**第1図**)。もっともシンプルな例では，メニューがなくムービーが一つで，ディスクを入れると再生が自動的に始められます。最後まで再生して停止させたり，ストップキーを押すまでリピート再生するようにもできます。チャプタを設けて前後へのジャンプも可能なようにできます。

また，複雑なものとしては市販の映画タイトルにあるように，最初に短時間のムービーを再生した後にメニューで止まり，そこからメインのムービーの全編再生やチャプタへのジャンプ，さらに別のメニューに飛んで別のムービーへというように作ることができます。目的に応じて使いやすく，見たいシーンがすぐに探せるDVDを作ることができます。

DVDitのワークスペース各部の名称

メインワークスペースはシンプルに見えますが，すべて手動で操作していきますので，多機能な操作を行うボタンが無駄なく機能的に配置されています(**画面11**)。「ビデオモニタ」の右側に，使用するメニュー/ムービーのリストが，サムネイル表示されます。その右外側にあるのがパレットウインドウで，ここにはムービー，メニュー背景，ボタンなどのすべての素材を置いておきます。ビデオモニターの下にムービータイムラインがあり，ここでチャプターポイントの設定などができます。

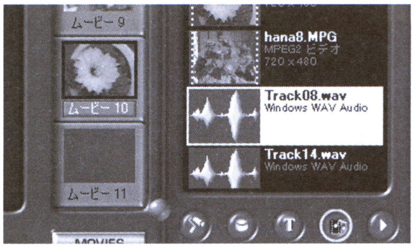
メディアファイルを準備

まず編集が完了したビデオを用意します。長いムービーにもチャプタを設定できますが，チャプタ単位で編集して保存しておいた方が作業性はよくなります。しかし，演奏会などのオーディオトラックを切れない所にチャプタが必要なときはDVDitで対応します。さらに，メニューで必要となる背景画像，テーマにあったボタン素材，オーディオ素材などを揃えます(**画面12,第1表**)。

ファーストプレイの作成

ファーストプレイは，ディスクが入ると自動的に指定したムービー，またはメニューが再生されるようにDVDプレーヤを動かす機能です。

背景，ビデオ，または静止画像をファーストプレイスフォルダにドラッグ・アンド・ドロップします。背景の場合はメニュー1，ビデオか静止画像だとムービー1として登録されます。ムービー/メニューリストの下にあるボタンで，ムービーリストとメニューリストを切り替えます。変更するときは新しい素材のサムネイルをドラッグ・アンド・ドロップして取り替えます。



《画面13》ムービーにサウンドトラックを追加

ムービーを登録

メニュー/ムービーボタンをムービー表示にします。メディアリストからビデオ，または画像のサムネイルを空のプレイスフォルダにドラッグ・アンド・ドロップしていきます。すでに表示されているフォルダに新しい素材のサムネイルをドラッグ・アンド・ドロップすると入れ替わります。プレイスフォルダには「ムービー3」というように，区別するために連番が付きますが，分かりやすい名前に変えることもできます。

ムービーにサウンドトラックを追加できます。メディアリストから，オーディオのサムネイルをムービープレイスフォルダにドラッグ・アンド・ドロップします。サウンドトラックの削除は，プレイスフォルダを右クリックして，ポップアップメニューから「オーディオをクリア」を選択します(**画面13**)。

チャプタポイントの設定

「DVDit」のSEバージョンはチャプタ機能をもっています。ムービーの任意のところにチャプタポイントを設定することができます。タイムライン上で作業します。

チャプタポイントを設定するタイムラインをダブルクリックします(**画面14**)。新しいチャプタポイントができ，プレイヘッドが移動してきて，チャプタポイントのフレームがメインウインドウに表示されます。

正確な位置に設定するには，右端にあるタイムコードの「T」ボタンをクリックして「C」に切り替えます。タイムコードの数値（時間：分：秒：フレーム数）を入力することにより，正確な位置にチャプタポイントを設定できます。


《画面14》チャプタポイントはタイムラインをダブルクリック，タイムコードの入力でも設定できる

チャプタポイントの移動はCtrlキーを押しながらドラッグします。また，Ctrlキーを押しながら矢印キーを押すと，1秒ずつ移動させることができます。

メニューを作成

メニューを作成するためには，背景を選択し，背景にボタンとテキストを追加し，次にムービーおよび他のメニューにボタンをリンクします。ボタンにはグラフィックス，テキストオブジェクト，ムービーやメニューのサムネイルを使うことができます。

①背景を決める

まずメニューリストを表示します。背景リストから空のプレイスフォルダにドラッグ・アンド・ドロップします。背景に使用する画像はアスペクト比4:3，解像度640×480(ピクセル)のものを使います。デジカメで撮った写真や，グラフィックソフトで作ったテクスチャなど，ファイル形式がサポートされていれば使用可能です。

②ボタンの追加

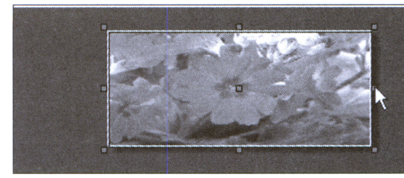
メニューには，他のムービーやメニューにリンクする，最大36個のボタンを作成することができます。また，グラフィックスやテキストオブジェクトからボタンを作ることができます。Photoshopなどで透明なパレットに作った図形をインポートしてボタンとすることができます。

また，メディアリストから，ビデオまたは画像のサムネイルをメニュー背景にドラッグ・アンド・ドロップすると，サムネイルが前面に表示されたボタンが作られます。同時にムービーリストにリンクされた状態で登録もされます。さらに，チャプタポイントにリンクしたボタンが作れます(次項で説明します)。

ボタンの大きさはマウスのドラッグで変更できます。また色などはツールバーの「効果」で変えることができます。

③テキストの追加

ボタンの識別や，メニューのタイトルなどの他に，テキストそのものにボタンと同じリンク機能をもたせることができます。テキストリストから使うフォントを，メニュー背景にドラッグ・アンド・ドロップします。「テキスト」または「Text」をダブルクリックして強調表示させて任意の文字を入力します。


《画面15》大きさ，縦横比，位置はマウスで変えられる



《画面16》グラフィックス，テキストなどがボタンとなる

メニュー画面への登録順にボタンとテキストに通し番号がつきます。DVDプレーヤーがメニューを表示するとリンクが設定してある最も小さい番号が付いているボタン，あるいはテキストが強調表示されます(**画面15,16**)。

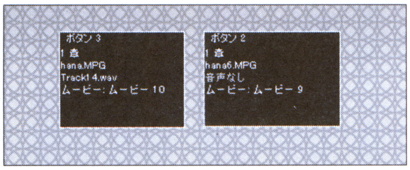
④サウンドトラックの追加

メニュ画面が表示されている間，繰り返し再生されるサウンドトラックを追加できます。メディアリストからオーディオのサムネイルをメニュープレイスホルダにドラッグ・アンド・ドロップします。

メニューボタンをムービーおよび他のメニューにリンク

メニューに配置したボタン，テキストオブジェクトを，ムービーや他のメニューにリンクします。すでにリンクが張られているビデオや画像などのボタンは，そのままにしておきます。メディアリストからボタン，あるいはテキストとリンクを張るビデオなどをドラッグ・アンド・ドロップします。他のメニューとリンクさせるには，メニュープレイスホルダのサムネイルをボタンにドラッグ・アンド・ドロップします。

チャプタポイントを設定したムービーとリンクしているボタンをクリックすると，タイムラインにチャプタポイントが



《画面17》リンク先が確認できる

表示されます。このチャプタポイントを背景にドラッグ・アンド・ドロップすれば，そのポイントのサムネイルが表示されたボタンが作られます。

リンクの確認は，メインウインドウで右クリックから「ボタンリンクを表示」を選ぶと，リンク先などの情報が分かります(画面17)。

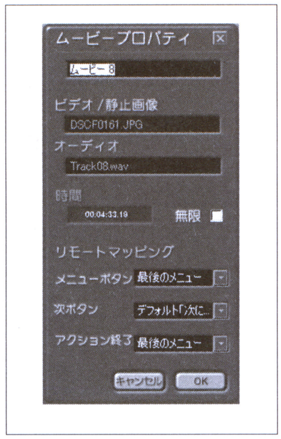
ムービーおよびメニューのプロパティの変更

プレイスホルダを右クリックしてプロパティを選ぶと，ダイアログウインドウが表示されます。ムービープロパティはDVDプレーヤがどのように動作するかを指示します。ムービーの再生が終わったとき，リモコンの「次へ」を押したときの動作などを設定します(画面18)。

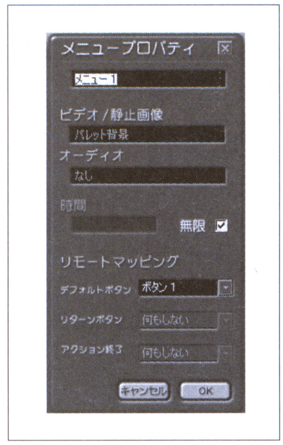
同じようにメニュープロパティのダイアログウインドウで，メニューの表示時間，リモコンの「戻る」を押したときの動作などを設定します(画面19)。

プロジェクトをプレビュー

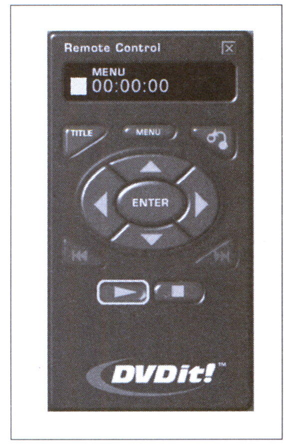
作成中に，設計どおり動作するかを確認できるDVDプレーヤが用意されています。リンクの確認やオーディオ，ビデオの確認をしながら作業を進めます。また，オーサリングがひととおり終わったら，



《画面18》DVDプレーヤの動作を指示するムービープロパティ



《画面19》メニューに関する動作をDVDプレーヤ指示するムービープロパティ



《画面20》リモコンを使用してDVDプレーヤで動作確認

徹底的にジャンプ，サーチなどが機能するか，画質・音質に問題はないかなどをチェックします(画面20)。

プロジェクトの終了

DVDを作る最後の工程です。編集結果を保存したときに，AVI形式で保存したか，DVD用ファイルのMPEG2形式で保存したかで処理時間が変わってきます。AVI形式だとMPEG2形式にエンコードする時間は，それなりにかかりますが，DVDitが計算して最高のビットレートを選んでくれます。

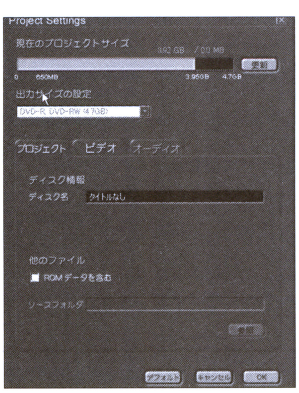
「ファイル」「プロジェクト設定」で「Project Setting」ダイアログを表示します。このダイアログを立ち上げるたびに，必ず「現在のプロジェクトサイズ」の「更新」をします。この容量表示バーが赤色になったらディスク容量を超えていますので，再度出力サイズなどの設定をし直します(画面21,22)。

DVDの作成

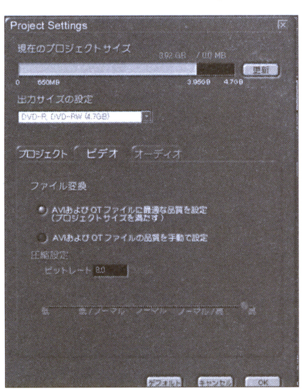
外付けドライブを使用している場合は，工程に入る前に接続して電源を入れておきます。ディスクを入れて，「ビルド」「DVDディスクの作成」を選択。DVDディスク作成のダイアログで書き込みに関する設定をします。「一般」と「詳細」の2面あります。始めのうちは時間はかかりますが，テスト書き込みをして確認します(画面23,24)。

テスト書き込みがうまくいったらプロジェクトの保存をしておきます。

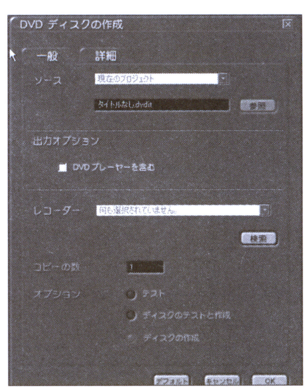
ここでも，リライタブルディスクで確認するほうがよいでしょう。また，最近売られている廉価なディスクの中には，品質の良くないものがありますので気をつけて下さい。特に長期保存や，ディスク面をフルに使う1時間以上のプロジェクトの書き込みには，信頼性の高い国内メーカーのものを使うようにします。



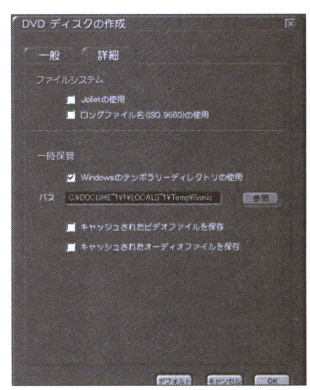
《画面21》ディスクのタイプ，プロジェクトの容量などを設定・確認する



《画面22》「ビデオ」ではビットレートが設定できる



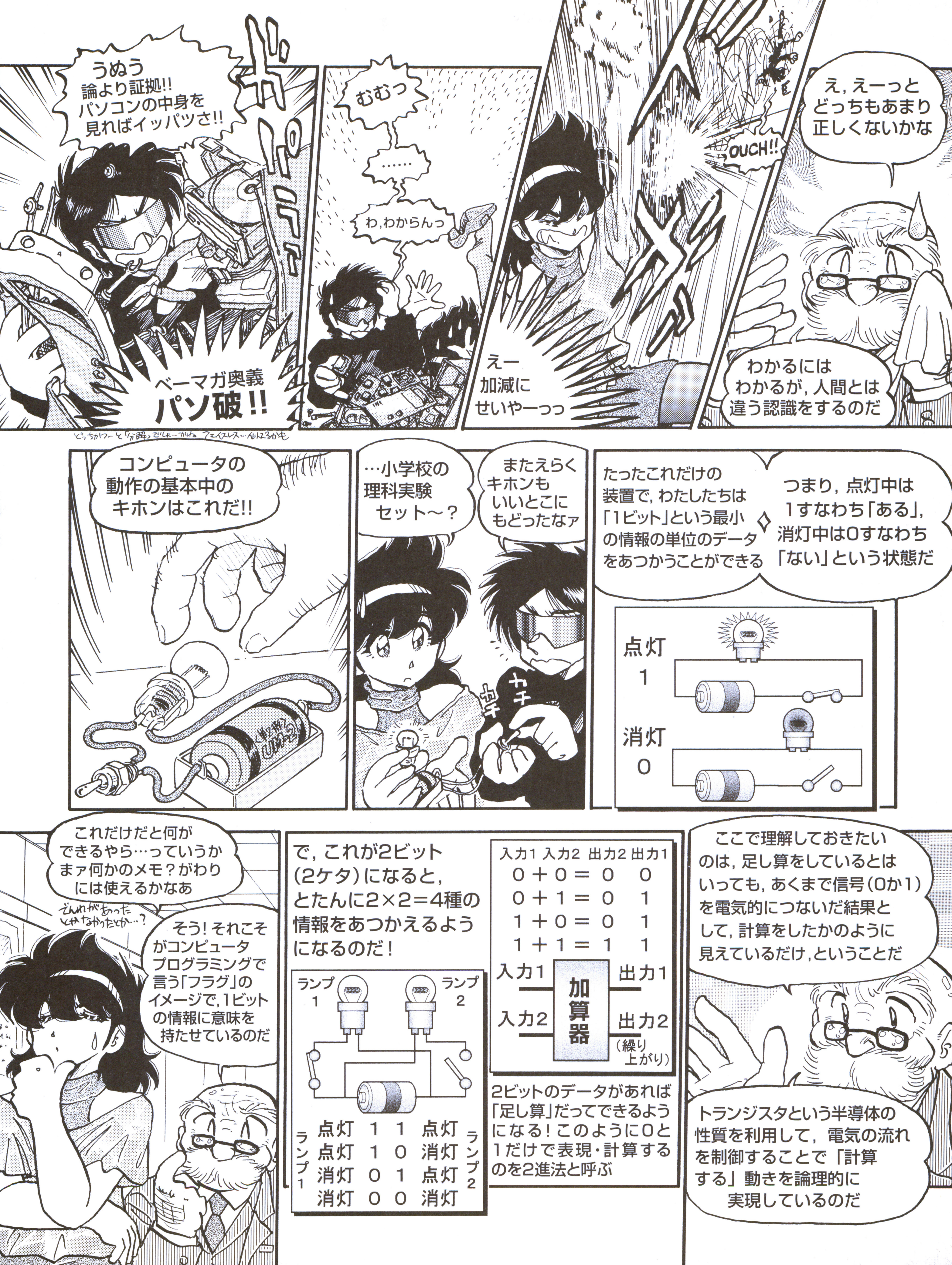
《画面23》レコーダーに関する書き込み条件の設定



《画面24》詳細ではテンポラリファイルなどの設定をする

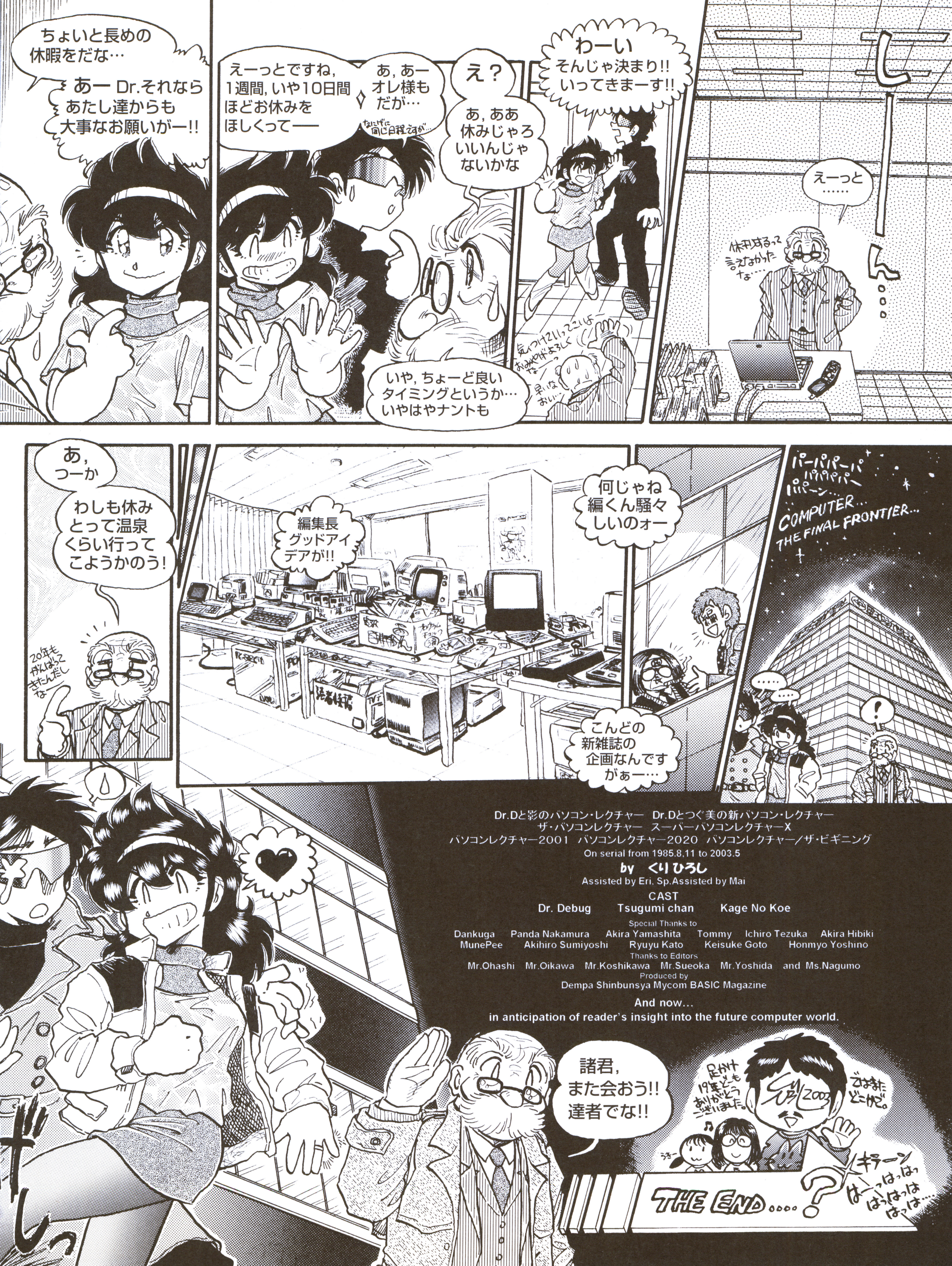


Dr.Dとつぐ美と影の
パソコンレクチャー
by
くりひろし
LAST STAGE:
真のプラグラミング
をめざせ！
超電脳黙示録に…レッツ！ファイナル！
アクセス承認!!
フュージョン!!!
電波新聞社
某月某日
DEMPAビル…
影くん
まじめに
やってるかな
BASIC
編集部
お？
ピー
あのさ～
ぎゃー
あーでもない
こーでもない
おらおら
だーかーらー
パソコンは人間の言葉
なんてわかんないから
プログラミング言語が
あるのよー
ちーがーうー
パソコンのCPUは電子
頭脳なので人間様のコトバ
くらい理解できるから
高級言語がー


うぬう
論より証拠!!
パソコンの中身を
見ればイッパツさ!!
ドパァァ
ベーマガ奥義
パソ破!!
むむっ
……
わ,わからんっ
OUCH!!
えー
加減に
せいやーっっ
え,えーっと
どっちもあまり
正しくないかな
わかるには
わかるが,人間とは
違う認識をするのだ
どっちかつーと「分解」でしょーがね。フェイスレス…似てるかも
コンピュータの
動作の基本中の
キホンはこれだ!!
…小学校の
理科実験
セット〜？
またえらく
キホンも
いいとこに
もどったなァ
カチ
カチ
たったこれだけの
装置で,わたしたちは
「1ビット」という最小
の情報の単位のデータ
をあつかうことができる
つまり,点灯中は
1すなわち「ある」,
消灯中は0すなわち
「ない」という状態だ
点灯
1
消灯
0
これだけだと何が
できるやら…っていうか
まァ何かのメモ？がわり
には使えるかなあ
それがあったとかなかったとか…？
そう!それこそ
がコンピュータ
プログラミングで
言う「フラグ」の
イメージで,1ビット
の情報に意味を
持たせているのだ
で,これが2ビット
(2ケタ)になると,
とたんに2×2=4種の
情報をあつかえるよう
になるのだ!
ランプ1
ランプ2
ランプ1
点灯 1 1 点灯
点灯 1 0 消灯
消灯 0 1 点灯
消灯 0 0 消灯
ランプ2
入力1 入力2 出力2 出力1
0 + 0 = 0 0
0 + 1 = 0 1
1 + 0 = 0 1
1 + 1 = 1 1
入力1
入力2
加算器
出力1
出力2
(繰り上がり)
2ビットのデータがあれば
「足し算」だってできるよう
になる!このように0と
1だけで表現・計算する
のを2進法と呼ぶ
ここで理解しておきたい
のは,足し算をしているとは
いっても,あくまで信号(0か1)
を電気的につないだ結果と
して,計算をしたかのように
見えているだけ,ということだ
トランジスタという半導体の
性質を利用して,電気の流れ
を制御することで「計算
する」動きを論理的に
実現しているのだ



そういう意味で，コンピュータはすべてが2進法の論理回路でできているのは本質であり必然だ!!
もし10進法のコンピュータがあったなら，100を表現するのに最低19個のランプがいる
百の位 十の位 一の位
2進法だったら，わずか8個のランプで100以上が表せる
128 64 32 16 8 4 2 1
つまりコンピュータは膨大な数の2進数スイッチの組み合わせの塊
実際，ごく初期のコンピュータプログラミングは回路の配線をつなぎかえてやっていたし…
若いころは良かったんじゃが…
ううう
2進数より少しは読みやすい16進数に置きかえたマシン語でプログラムした時代もあった
（なぜ16進かというと，2の倍数では10よりも16の方がキリがいい）
現在のプログラミング言語というのは，わかりにくい2進法からだんだん「人間に理解しやすく」まとめていった結果なのだよ したがって…
高級言語
input A=7;
PUSH DX, 7
アセンブラ言語
5BE9
16進数（マシン語）
0101110111101001
2進データ
BASICやCなどの言語は人間に理解できるコトバである一方，コンピュータが実行するための信号のら列に置きかえられるものなのだ
そうか，プログラムがわかりやすいとかわかりにくいとか…それは，プログラムを書く人間のせいだったのね
BASIC＝ホビー向け
C＝OS向け
C++＝オブジェクト指向
VB＝GUI向け
JAVA＝ポータブル
これらは例えですが…
ある一面じゃな
そのとおり! プログラム言語はそれぞれ「何の目的で」があるから，向いていない言語ではプログラミングしづらいことになる
そして，プログラムの構造自体も3月号で少しやったプログラムの3大基本制御の組み合わせですべて解決している
《順次処理》
処理1 → 処理2 → 処理3
《分岐処理》
処理1 → 判断 → 処理2 / 処理3
《繰返し処理》
処理1 → 判断 → 処理2
Windowsだろうが Linuxだろうが，世の中のあらゆるプログラムは0と1の繰り返しだし，3大基本制御だけで書かれていると言っても過言ではない
でもまあ，一見してそんなものの組み合わせに見えないわ
ムヅカシイことを知らなくてもカンタンなプログラムは作れるし，作らなくたって何でもソフトはあるし…パソコンも進化したってコトなのかしら？
ベーマガ創刊当時は，パソコンも8ビットプロセッサで処理速度も4MHzとか小さかった…今のコンピュータの1/1000だな
Dr. Dからのおねがい
比べちゃいかんのだろうけど，これから次世代の電脳界を担うベーマガ読者諸君には，「コンピュータはなぜ動いているか？」を知ったうえでプログラミングをしてみてほしい! 言語をおぼえただけでは真のプログラミング技術～ベーマガの奥義は身につかないと知ってほしい!!
あ，ところで創刊250号を超えた記念ってわけでもないんじゃが実は大事な話があるのだが…
ひとだんらくついたので…
はい？


ちょいと長めの休暇をだな…
あー Dr.それなら あたし達からも大事なお願いがー!!
えーっとですね, 1週間, いや10日間ほどお休みをほしくって──
あ, あー オレ様もだが…
え?
あ, ああ 休みじゃろ いいんじゃないかな
いや, ちょーど良いタイミングというか… いやはやナントも
わーい そんじゃ決まり!! いってきまーす!!
し──ん…
えーっと ……
休刊するって言えなかったな…
あ, つーか
わしも休みとって温泉くらい行ってこようかのう!
20年もがんばってきたんだしな~
編集長グッドアイデアが!!
何じゃね 編くん騒々しいのォー
こんどの新雑誌の企画なんですがぁー…
パーパーパーパ パパパパー パーン
COMPUTER… THE FINAL FRONTIER…
!
Dr.Dと影のパソコン・レクチャー Dr.Dとつぐ美の新パソコン・レクチャー
ザ・パソコンレクチャー スーパーパソコンレクチャーX
パソコンレクチャー2001 パソコンレクチャー2020 パソコンレクチャー/ザ・ビギニング
On serial from 1985.8,11 to 2003.5
by くりひろし
Assisted by Eri, Sp.Assisted by Mai
CAST
Dr. Debug Tsugumi chan Kage No Koe
Special Thanks to
Dankuga Panda Nakamura Akira Yamashita Tommy Ichiro Tezuka Akira Hibiki
MunePee Akihiro Sumiyoshi Ryuyu Kato Keisuke Goto Honmyo Yoshino
Thanks to Editors
Mr.Ohashi Mr.Oikawa Mr.Koshikawa Mr.Sueoka Mr.Yoshida and Ms.Nagumo
Produced by
Dempa Shinbunsya Mycom BASIC Magazine
And now…
in anticipation of reader's insight into the future computer world.
諸君, また会おう!! 達者でな!!
ではまたどこかで。
ギラーン
はーっはっはっ はっはっは はっは…
THE END….?
ザしっ

協力：（株）富士通南九州システムエンジニアリング

はじめよう！ Microsoft Officeで BASICプログラミング 特別号

Windows 意外と知らない便利なテクニック

●文/椛島たけし

ベーマガ休刊。私もびっくりしました。最終回記念？
というわけで，突然ですが便利なWindowsの操作テクニックを紹介することにしましょう。

ちょっとした操作方法

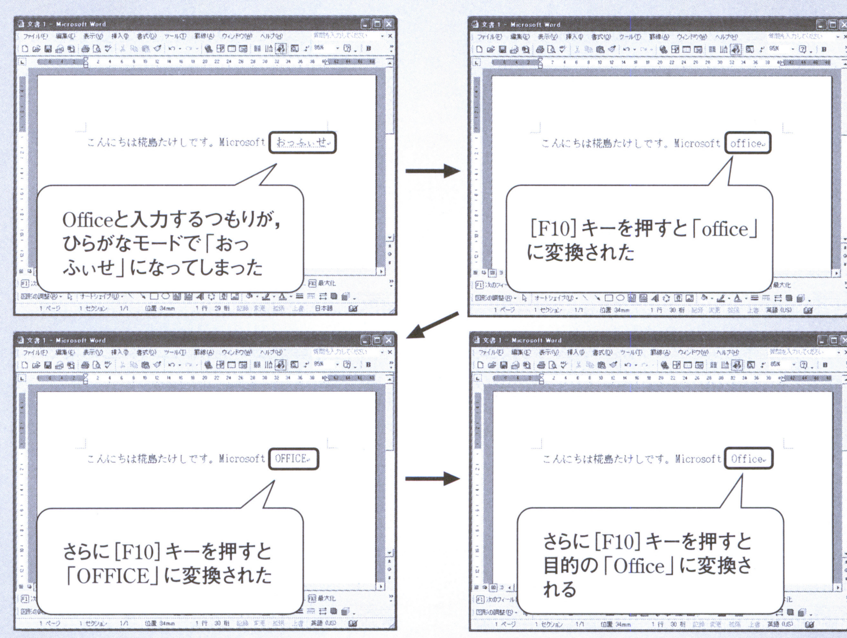
私の周りでは，毎日何人もの人がWindowsに向き合っています。ExcelやWordで資料を作ったり，VBやJavaでプログラミングをしたりと，やってることは微妙に違うのですが，みんなマウスやキーボードを軽快に操っています。ちなみに仕事が速い人ほど，キーボードからほとんど手が離れず，マウスをあまり使っていません。

ちょっと眺めていると……な，何？いまの操作？なんでIME2000で[F10]を使って変換しているの！？それって技？疑問は深まるばかりです。というわけで，Officeに限らずWindowsを便利に使うコツを紹介してみましょう。

魔法のキー [Ctrl]

まずは入門編。コピー，貼り付け，切り取りを，まだマウスでやってませんか？[Ctrl]を左手の小指で自在に押せるようになると，とても操作が楽になりますよ。「元に戻す」操作は結構穴場ですね。

選択範囲のコピー [Ctrl]＋[C]
選択範囲の切り取り [Ctrl]＋[X]
[Ctrl]＋[Del]
貼り付け [Ctrl]＋[C]
[Ctrl]＋[Ins]



▲《第1図》 [F10]でひらがなからアルファベットに変換

すべてを選択　Ctrl＋A
元に戻す　Ctrl＋Z

保存もキーボードが当たり前

突然の電源断や原因不明のプログラムのフリーズなどが発生しても慌てないためには，作成したプログラムや文書などをこまめに保存することが大事ですね。マウスでいちいち[ファイル]-[保存]を選ぶのも面倒です。そんな時にはキーボードから一発保存。万が一に備えましょう。

上書き保存　Ctrl＋S

F10キーでアルファベット変換

「Office」と入力しようとして，「おっふぃせ」となってしまうケースって結構ありますね。Microsoft IMEで，ひらがなモードでアルファベットを入力すると，せっかく入力したアルファベットがひらがなになってしまいます。そんな時は消さずにF10でアルファベットに変換しましょう(**第1図**)。

この変換の優れているのは，「すべて小文字」「すべて大文字」「先頭だけ大文字」と順に変換してくれることです。英単語と日本語が混じった文章を書くときでも，ず〜っとひらがなモードでOKです。

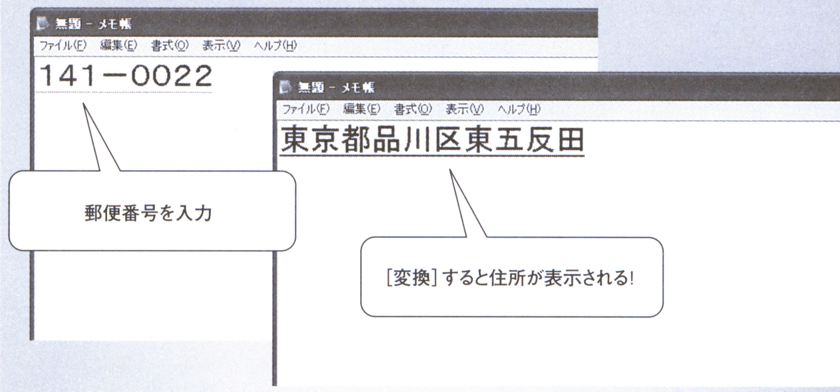
住所をいちいち入力していませんか?

Microsoft IMEでは，郵便番号の辞書をもっていますので，郵便番号を入力して変換すると住所に変換してくれます(**第2図**)。住所録の編集や，Webでユーザ登録をする時などとても便利ですね。"ー"を忘れずに入力するのがポイントです。

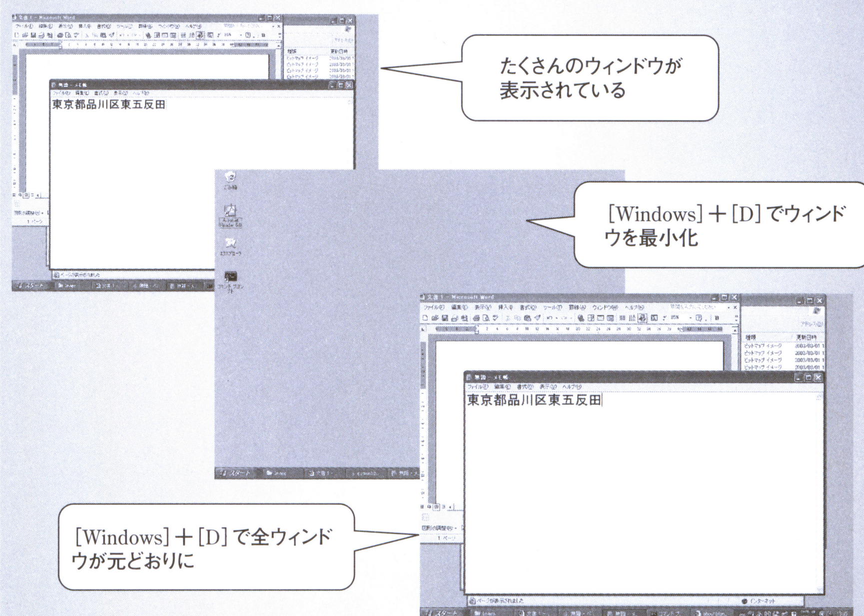
ただし，個別の郵便番号(企業などに個別に割り振られている番号)は変換できませんので注意してください。

Alt＋半角/全角はもう古い?

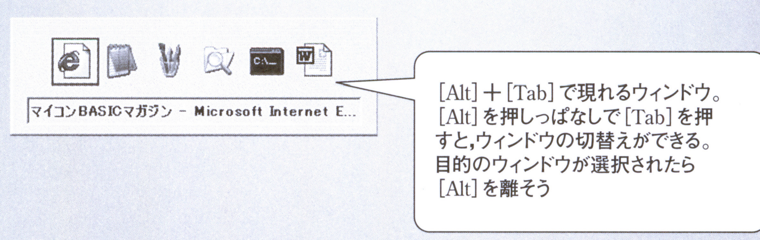
Microsoft IMEを起動して日本語を入力したいとき，Alt＋半角/全角で



▲《第2図》 Microsoft IMEで郵便番号変換



▲《第3図》 すべてのウィンドウを最小化する



▲《第4図》 [Alt]+[Tab]でウィンドウを選択

IME2002を起動している人も多いと思います。特に，MS-DOS時代からパソコンを使っている人は身についている操作ですね。でも，IME2002などでは半角/全角だけで起動します。後生大事にAltを押していたのが無駄に思えてきます……。

でも，コマンドプロンプトや他のIMEではAlt＋半角/全角で起動する必要があります。

ウィンドウの終了もキーボードから

ウィンドウを終了させるにはAlt＋F4を使用します。すべてのウィンドウを終了させたあとにAlt＋F4を押すとコンピュータの電源を切る画面が表示されます。

全部のウィンドウを一気に最小化する

デスクトップ上のアイコンをクリックしたいのに，ウィンドウが邪魔でしょうがない場合って結構ありますよね。そんな時にはWindows＋M，すべてのウィンドウが一気に最小化されます。反対に，最小化されたすべてのウィンドウをすべて戻すにはWindows＋Shift＋Mです。

ちなみに，どちらの操作もWindows＋DでもOKです(**第3図**)。

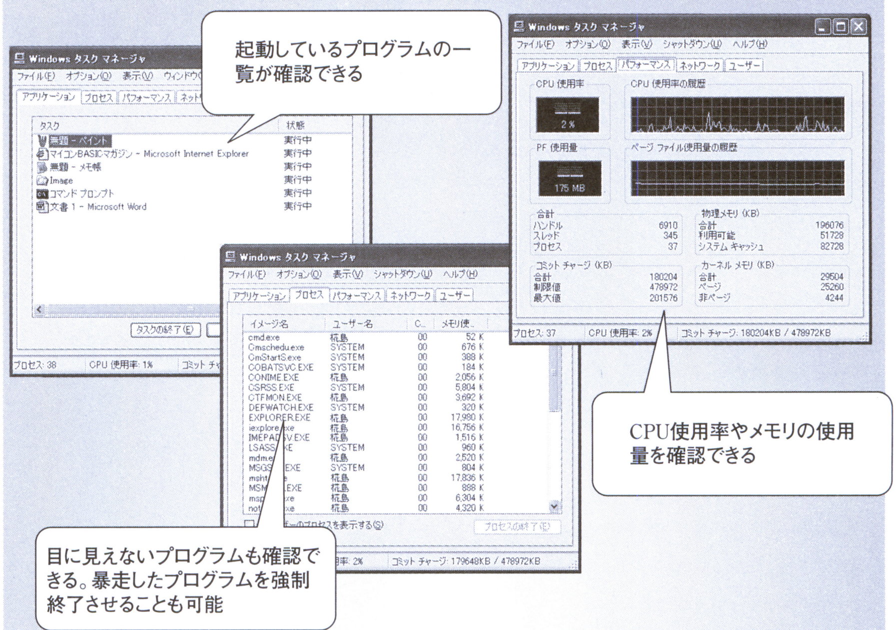
Alt＋Tabで選択

ウィンドウが沢山表示されている状態で，使いたいウィンドウをすばやく前面に表示する方法がAlt＋Tabです(**第4図**)。

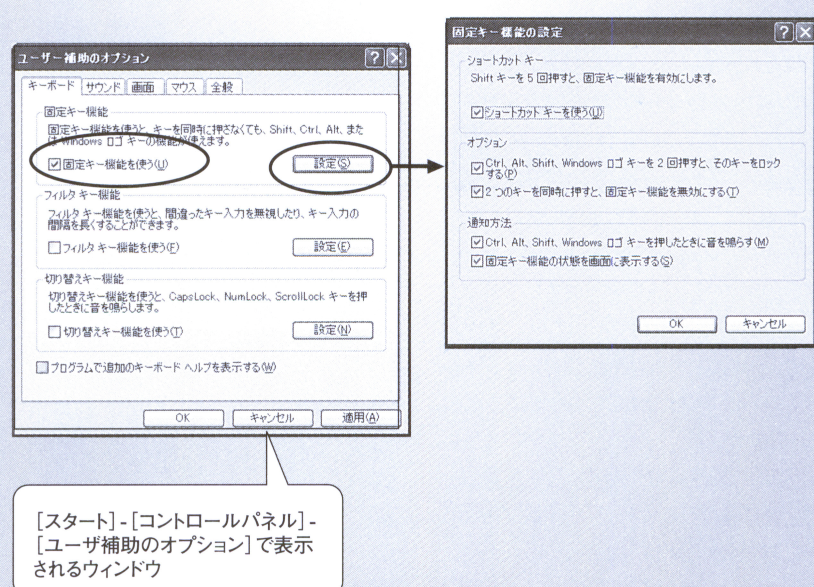
Altを押しっぱなしでTabを押すたびにウィンドウの切替ができますので，すばやく操作できますね。

HomeキーとEndキー

行などの先頭に戻るのがHome，最後に移動するのがEnd。この二つもかなり使い勝手が良い操作です。また，Ctrl＋Homeとするとページなどの先頭，Ctrl＋Endでページなどの最後に移動します。



▲《第5図》 [Ctrl]+[Shift]+[Esc]でタスク マネージャを起動しよう



▲《第6図》 固定キーを使うと同時にキーを押す必要がなくなる

アクティブなウィンドウをコピー

プログラムの操作説明書などをつくるとき，画面を取り込む必要があります。[PrintScreen]で画面全体がコピーされますが，[Alt]＋[PrintScreen]でアクティブな画面だけをコピーすることができます。

タスクマネージャを上手に使おう

暴走したプログラムを強制停止させたりするときに便利なタスクマネージャ(**第5図**)ですが，一発で起動する方法があります。

タスクマネージャの起動

[Ctrl]＋[Shift]＋[Esc]

指一本でキーを同時に押す？

二つ以上のキーを同時に押すテクニックを中心に紹介しましたが，「ハンバーガーを食べながら」「パソコン雑誌を見ながら」など，ながら操作のとき，二つ以上のキーを同時に押すのは，ちょっと面倒な場合があります。そんな時には[ユーザ補助]の[固定キー]にチェックを入れておくと便利です(**第6図**)。

Internet Expolorerで入力を無かったことに

Webアクセスで掲示板などに書き込んでいる時，書込み自体を最初からやり直したいケースもあります。Internet Explorerの[戻る]で戻るのも再読込に時間がかかるし……。そんな時には[Esc]キーで入力内容がすべて消えます(**第7図**)。

ホームポジションが大事

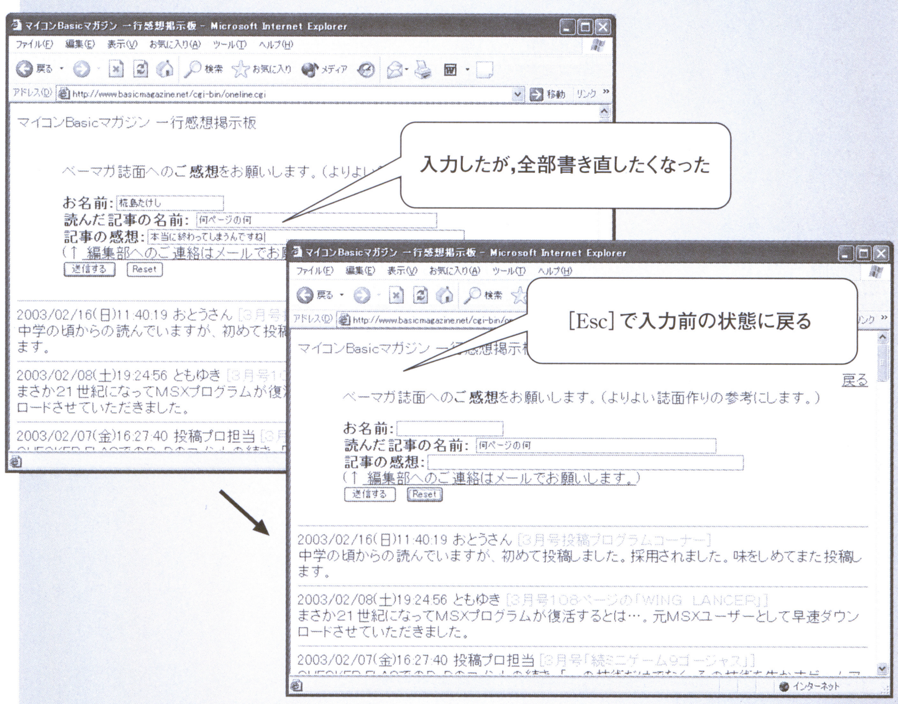
今回の操作の多くは，キーボードの左手前キーを利用しています。右手はマウスで左手はキーホードという状況は結構ありますので，左手を上手に使いこなすことがスムーズな操作のポイントです。

左手の人差し指を[F]，小指を[A]に置いておくホームポジションのクセをつけると次第に操作がスムーズになると思います。とりあえず，左手を正しく置くことが大事ですね。

最後に

みなさん長い間，椛島たけしの連載にお付き合いいただきありがとうございました。「F-BASIC」時代からExcel，Officeと，BASICに関する連載を4年半，計55回もさせていただきました。

休刊ということで，入門者から上級者まで，幅広い読者にプログラムの楽しさを伝える雑誌は事実上なくなったと思います。しかし，ベーマガはパソコンの黎明期から現在にいたるまで，コンピュータへの登竜門として多くの人に愛され，多くの人材を生み，日本のコンピュータ業界の発展の一翼を担った雑誌であったと感じています。そのような雑誌に参加させていただき，読者のみなさんおよびベーマガ関係者のみなさん，大変ありがとうございました。



▲《第7図」 Internet Explorerで入力した情報を取り消す

教科「情報」など授業にも使える！

Visual Basic アルゴリズム講座

文・プログラム
／東 幸太

第14回 オブジェクト指向の考え方

プログラムを作成するスタイルとして，オブジェクト指向というものがあります。通常，プログラムを作成していく場合はプログラム自体が作成対象となっていきますが，オブジェクト指向では機能ブロック単位にプログラムとデータを統合し，作成した機能ブロックが部品のように使えるために，必要となる要素を作成していきます。
今回は，このオブジェクト指向の考え方と，Visual-Basicでの例を説明します。

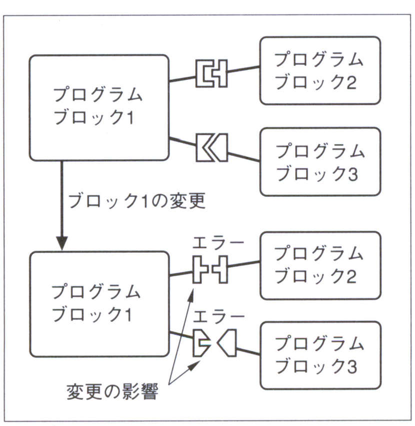


衝撃のBASICマガジン休刊。初心者向けプログラミング雑誌がまた一つ減ってしまう。情報量ではすでにWebが圧倒しているものの，意識しないとリストが見られないことの多いWebと，無意識にリストが見られる書籍では効果が大違いなのだが。

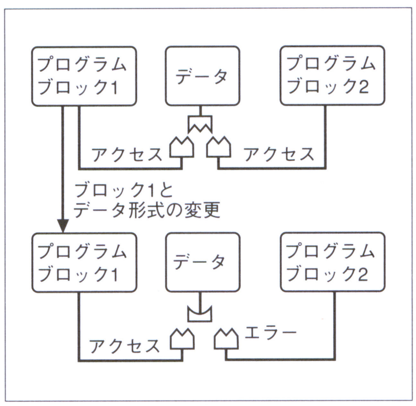
プログラム開発手法の歴史

プログラミングが可能なコンピュータが登場してから，プログラムはコンピュータを動かすために欠かせない要素となりました。初期のプログラム言語では，単純なプログラミングスタイルがとられており，必要な機能を単純に並べるプログラミングが行われてきました。

ところが，コンピュータの性能が向上するにつれ，プログラムに要求される機能は増加し，プログラムサイズもどんどん大きくなっていきました。この結果，作成したプログラムを後から修正・変更しようとしても，修正・変更部分が他の部分に及ぼす影響を正確に把握することができなくなるという問題が生じるようになりました（**第1図**）。たとえ，本人が修正・変更する場合でも，プログラム作成からの経過時間が半年，1年ともなると，本人でも修正・変更の影響を正確には見積もれなくなるものです。これが他人ともなると，資料があったとしても修正・変更は容易ではありません。



《第1図》プログラム規模が大きいと変更の影響を把握しにくくなる



《第2図》プログラムが分離できてもデータ形式が相互に関係する

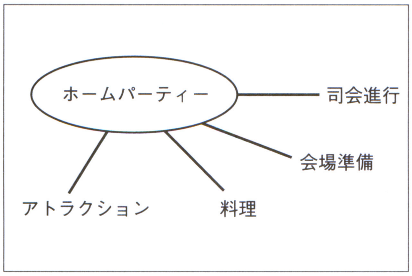
この問題に対する対策として，構造化プログラミングという手法が登場しました。構造化プログラミングでは，プログラムを機能ブロックごとに分離し，それぞれの機能ブロックの相互依存関係を少なくすることで，修正・変更の影響範囲を分かりやすくしています。機能ブロックのサイズを小さくしておけば，機能ブロック全体を見渡すことが容易になり，全体を把握した修正・変更が行えるようになります。

構造化プログラミングによって，ある程度は問題は解決しましたが，異なる機能ブロック間で共有するデータが存在する場合，このデータを経由して機能ブロック間が相互関係をもつようになります。この場合，機能ブロックＡでのデータアクセス手順の変更が，同じデータを扱う機能ブロックＢの処理に影響を及ぼす場合があります。こうなると，機能ブロックＡの変更を行う場合には，機能ブロックＢの内容も把握する必要が生じしてしまいます（**第2図**）。

このような依存関係は，個々のプログラムが同じデータを処理する上では避けられないものです。オブジェクト指向プログラミングでは，プログラムやデータなどの相互依存関係を最初に洗い出し，プログラムを作成するのと同時に，相互依存している部分をつないでいく手順でプログラムを作成します。

オブジェクト指向のイメージ

オブジェクト指向のポイントは，プログラム作成前に全体の相互関係を明確にし，設計を行うことにあります。設計段階では，プログラムやデータ，入出力といった要素は意識せずに，そのブロックが何を担当するのか，といった観点でブロックを作成していきます。ちょうど，複数名で何らかの作業を行う際の役割



《第3図》ホームパーティーの役割分担をイメージする

分担を決めるようなイメージになります。それぞれの役割がどのようにしてその役目を実行するか，といったことは意識せず，作業内容を各自に割り当てていきます。

例えば，ホームパーティーを開く場合を考えてみましょう。役割を考えてみると，**第3図**のように，司会進行，会場準備，料理，アトラクションというような内容が思いつきます。これらを複数名で分担する場合は，それぞれが役割になります。このおおまかな役割をどのようにして実現していくかを考えていくと，**第4図**のように展開されていきます。大きな役割分担が細かくなっていきます。ここで重要なのは，細かくした役割が簡単に実行できるようにすることです。簡単に実行できるようにするためには，個々のブロックの役割内容を減らすことが重要ですが，それ以外に役割を実行する際に他のブロックとのやりとりが少なくても済むようにすることがポイントになります。ホームパーティー全体の進捗状況を知らないと実行できない作業というのは，作業自体は簡単であっても情報を集めるために必要な労力が増えてしまい，結果的に大変な作業となってしまいます。

例えば，プログラム作成のブロックは，会場の情報，料理の情報，アトラクションの情報の大半を知らなければプログラムを作成できません。作業自体はプログラムのワープロ打ちをする程度ですが，情報収集にかかる労力はかなりのものになることでしょう。これに対し，料理の材料買い出しは，材料の種類や量によってはかなりの肉体労働になりますが，考えるべき範囲はメニュー決定ブロックから受け取った買い物リストだけで済みます。

役割分担が展開できたら，次はブロック間でやりとりすべき情報やモノを考えます。大抵の場合，あるブロックは別のブロックによる指示やモノを受け，さらに別のブロックに対して指示やモノを与えるような形になります。このようにブロック間を流れる情報やモノの経路を明確にするわけです。ホームパーティーの場合は**第5図**のようになります。

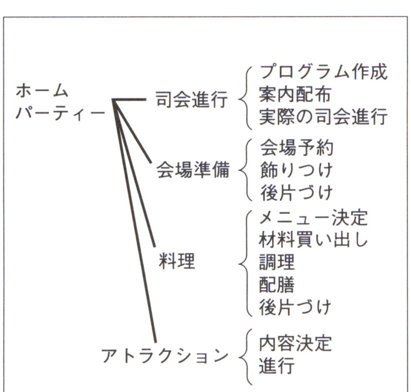
これでホームパーティーの設計ができました。あとは，個々のブロックを実際に担当する人を決めてやれば，設計情報によって情報やモノを誰から受け取り，どういう作業をして，誰に新しい情報やモノを送れば良いかがわかります。ここで重要なのは，設計を行ったことにより，個々のブロックを担当する上で，ホームパーティー全体を意識する必要が無くなっているということです。個々の担当者は自分の役割だけを設計どおりに実行すれば，全体としてホームパーティーが開催できるようになるわけです。

オブジェクト指向プログラミングでは，プログラムについてこのような設計を行い，個々のブロックが担当すべき処理やデータをはっきりさせるのと同時に，個々のブロックのプログラムを作成する際に把握しなければならない他のブロックなどの要素を極力少なくします。個々のブロックには，処理を行うプログラムに加えて，そのブロックで必要となるデータを用意します。うまく設計できれば，ブロックのプログラムを作成する時に必要となる，他のブロックのプログラムやデータなどの量が減り，見渡すべき範囲を小さくすることができます。一度に考えるべき要素が少なくなれば，プログラムは簡単に作れるというわけです。

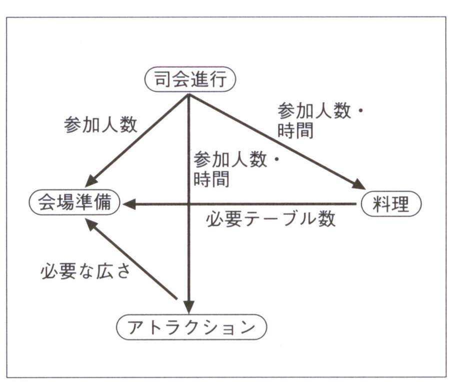
プログラム作成を目的とするプログラミングでは，ちょうどホームパーティーを一人で開催しようとするような形になります。すると，パーティーの会場掃除をしている間に料理の内容が気になったり，アトラクションの内容を考えている際にはパーティープログラムの作成が影響を受けたりします。プログラムを構造化し，それぞれの機能ブロックに分割すれば，構造化プログラミングとなってプログラムの内容面ではそれぞれの要素を分離することができます。しかし，取り扱うデータについては分離されず，データをとおして他の機能ブロックが影響をおよぼしてきます。

オブジェクト指向プログラミングの場合は，ホームパーティーの役割分担を行うように，プログラム作成前にプログラムの機能ブロックを分割するのと同時に，どの機能ブロックとどの機能ブロックがどういう関係をもつかをあらかじめ設計します。設計が完了したら，個々の機能ブロックを作成していきますが，その際には他の機能ブロックの影響はあらかじめ設計で明らかになった要素だけになり，他の機能ブロックの影響が，思いも寄らないところに現れるといったことが避けられます。

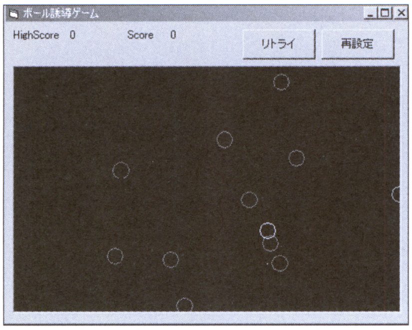
結果としてできあがるプログラムは，構造化プログラミングを行えば機能ブロックごとに分割されているという点ではあまり差が生じません。しかし，オブジェクト指向プログラミングによる設計を行ってあれば，個々の機能ブロックが取り扱うデータが，他の機能ブロックと関係しているのかしていないのか，関係しているとすればどのように関係しているか，といったことが，常に明確になります。



《第4図》ホームパーティーの作業分担



《第5図》ブロック間を流れる情報やモノを明確にする



《第6図》ボール誘導ゲーム。ボールはマウスカーソルまでの距離が遠いほど加速度が大きくなる。障害物の配置からルートを決め，右方向へはなるべく加速するだけにし，上下方向をうまく調整して時間を削っていこう

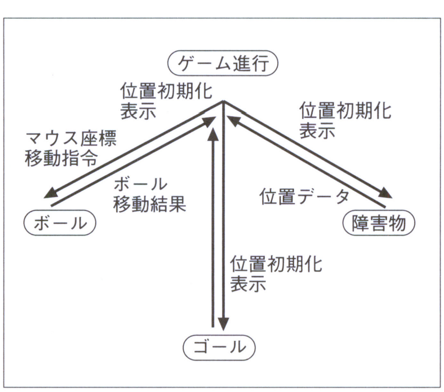
ボール誘導ゲーム

それでは，オブジェクト指向プログラミングの設計手法を使ったゲームを見てみましょう。**リスト1**はボール誘導ゲームです。**リスト1**を実行すると，**第6図**のような画面が表示されます。画面左にはボールが白で，右にはゴールが緑で，それぞれ表示されています。また，画面中には赤の障害物が存在します。マウスを左クリックするとゲームがスタートし，ボールはマウスカーソルの方向へ向かって加速し始めます（マウスカーソルはピクチャボックス範囲内に存在する必要があります）。マウスカーソルをうまく調整し，障害物を避けてボールをゴールに入れてください。ゴールに入るまでの時間に応じて得点が決まり，より速く入れるほど高得点になります。ボールが障害物に当たってしまうと得点は0です。

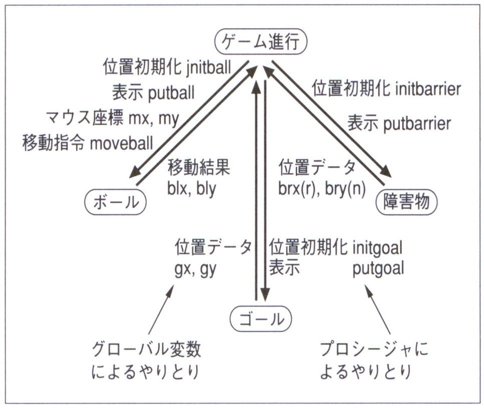
このリストでは，画面に登場する要素として，ボール，障害物，ゴール，そしてゲーム進行の四つをオブジェクトと見なしています。ゲーム進行オブジェクトを最も上位のオブジェクトとし，このオブジェクトが他のオブジェクトに指示を出していく形にします。それぞれの関係を**第7図**のように決め，相互に影響を与える要素を洗い出します。これを元に，影響をどのような手順で与えるようにするかを**第8図**のように決めていきます。

この設計を元にして**リスト1**は作成されています。相互にデータをやりとりする際にはグローバル変数を利用していますが，それらはどこからどのようにアクセスされるかが設計図によって明確になっています。よって，これらの変数に対する変更を行う際には，どこに注意すれば良いかがわかりやすくなっています。

リスト1では，障害物は移動しませんが，これらを少しずつ移動させるようにしてみてください。障害物のオブジェクト部分だけを変更すれば，障害物が移動するようになり，他の部分には影響を与えなくても済むことがわかります。



《第7図》リスト1のオブジェクトとオブジェクト間のやりとり



《第8図》オブジェクト間のやりとりをプロシージャ，グローバル変数で実現

リスト1の変数表

hiscore	最高得点
score	得点
xsize	ピクチャボックスの X方向サイズ
ysize	ピクチャボックスの Y方向サイズ
dt	刻み時間
g	加速係数
gf	ゲーム進行フラグ
mx	マウスポインタX座標
my	マウスポインタY座標
blr	ボール半径
blx	ボールX座標
bly	ボールY座標
blvx	ボールX方向速度
blvy	ボールY方向速度
brr	障害物半径
brx(9)	障害物X座標
bry(9)	障害物Y座標
gr	ゴール半径
gx	ゴールX座標
gy	ゴールY座標

終わりに

オブジェクト指向プログラミングは，最初の設計が重要です。設計段階で，ブロック間のやりとりが少なくなるような切り分けを行い，ブロックの独立性を高めてやるところにポイントがあります。独立性が高まれば，意識すべき範囲も限定され，ブロック間のやりとりが少なくなれば，プログラム全体の見通しがよくなります。急がば回れ，一度に見渡しきれない規模のプログラミングを行う際には，十分な設計をしてみてください。なお，今回は，オブジェクト指向用の記述方法などには触れていませんが，VBにはクラスファイルなどの機能がありますので，各自調べてみてください。

《リスト1》ボール誘導ゲームのプログラム・リスト

```
VERSION 5.00
Begin VB.Form Form1
   Caption         =   "ボール誘導ゲーム"
   ClientHeight    =   4956
   ClientLeft      =   48
   ClientTop       =   324
   ClientWidth     =   6768
   LinkTopic       =   "Form1"
   ScaleHeight     =   4956
   ScaleWidth      =   6768
   StartUpPosition =   3  'Windows の既定値
   Begin VB.CommandButton Command1
      Caption         =   "リトライ"
      Height          =   492
      Left            =   3960
      TabIndex        =   6
      Top             =   120
      Width           =   1212
   End
   Begin VB.Timer Timer1
      Interval        =   30
      Left            =   3480
      Top             =   120
   End
   Begin VB.CommandButton Command2
      Caption         =   "再設定"
      Height          =   492
      Left            =   5280
      TabIndex        =   5
      Top             =   120
      Width           =   1212
   End
   Begin VB.PictureBox Picture1
      Height          =   4092
      Left            =   120
      ScaleHeight     =   4044
      ScaleWidth      =   6444
      TabIndex        =   0
      Top             =   720
      Width           =   6492
   End
   Begin VB.Label Label4
      Height          =   372
      Left            =   2760
```

```
      TabIndex        =   4
      Top             =   120
      Width           =   852
   End
   Begin VB.Label Label3
      Caption         =   "Score"
      Height          =   372
      Left            =   2040
      TabIndex        =   3
      Top             =   120
      Width           =   612
   End
   Begin VB.Label Label2
      Height          =   372
      Left            =   1080
      TabIndex        =   2
      Top             =   120
      Width           =   852
   End
   Begin VB.Label Label1
      Caption         =   "HighScore"
      Height          =   372
      Left            =   120
      TabIndex        =   1
      Top             =   120
      Width           =   852
   End
End
Attribute VB_Name = "Form1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

'ゲーム進行関連変数
Dim xsize!, ysize! 'ピクチャボックスのサイズ
Dim hiscore% 'ハイスコア
Dim score% 'スコア
Dim dt! '刻み時間
Dim g! '加速係数
Dim gf% 'ゲーム進行フラグ
Dim mx!, my! 'マウスポインタ座標

'ボール関連変数
Dim blr!, blx!, bly!, blvx!, blvy! 'ボール半径と
座標・速度

'障害物関連変数
Dim brr! '障害物半径
Dim brx!(9), bry!(9) '障害物座標

'ゴール関連変数
Dim gr!, gx!, gy! 'ゴール半径と座標

'ゲーム進行関連プロシージャ

Private Sub Command1_Click() 'リトライ
    initgame2
End Sub

Private Sub Command2_Click() '再設定
    initgame
End Sub

Private Sub Form_Load()
    xsize = Picture1.Width 'ピクチャボックスのサ
イズを取得
    ysize = Picture1.Height
    hiscore = 0 '得点初期化
    initgame
End Sub

Sub initgame() 'ハイスコア以外初期化
    gf = 0 'ゲーム状況フラグ初期化
    dt = 0.1 '刻み時間設定
    g = 100 '加速係数設定
    score = 1000
    initball 'ボール初期化
```

```
    initbarrier '障害物初期化
    initgoal 'ゴール初期化
    initfield 'ゲームフィールドクリア
End Sub

Sub initgame2() 'リトライ時の初期化
    gf = 0 'ゲーム状況フラグ初期化
    score = 1000
    initball 'ボール初期化
    initfield 'ゲームフィールドクリア
End Sub

Private Sub Picture1_Click() 'マウスクリック
    If gf = 0 Then
        gf = 1 'ゲームスタート
        initball
        initfield
    End If
End Sub

Private Sub Timer1_Timer() '一定時間間隔で呼び出さ
れるタイマ
    If gf = 1 Then
        If score > 0 Then score = score - 1 'スコ
アを減らしていく
        putscore
        moveball
        check
        If gf = 2 Then 'ゴールしたか？
            If score > hiscore Then hiscore =
score 'ハイスコア更新
            putscore
            puthighscore
        ElseIf gf = 3 Then 'ミスしたか？
            score = 0
            putscore
        End If
    End If
End Sub

Sub check() '各種接触判断
    Dim i%
    '画面外へ出たか？
    If blx < 0 Or bly < 0 Or blx > xsize Or bly >
ysize Then gf = 3
    'ゴールしたか？
    If (blx - gx) * (blx - gx) + (bly - gy) * (bly
- gy) < (blr + gr) * (blr + gr) Then gf = 2
    '障害物に接触したか？
    For i = 0 To 9
        If (blx - brx(i)) * (blx - brx(i)) + (bly
- bry(i)) * (bly - bry(i)) < (blr + brr) * (blr +
brr) Then gf = 3
    Next
End Sub

Sub picture1_MouseMove(b%, s%, x!, y!)
    mx = x 'マウスポインタの位置を保存
    my = y
End Sub

Sub initfield() '画面初期化
    Picture1.AutoRedraw = True
    Picture1.Line (0, 0)-(xsize, ysize), 0, BF
    putball 1
    putbarrier 1
    putgoal 1
    putscore
    puthighscore
End Sub

Sub putscore()
    Label4.Caption = Format$(score, "####0")
End Sub

Sub puthighscore()
    Label2.Caption = Format$(hiscore, "####0")
End Sub
```

```
'ボール関連担当プロシージャ

Sub initball() 'ボール初期化
    blr = xsize / 50
    blx = blr / 2
    bly = ysize / 2
    blvx = 0
    blvy = 0
End Sub

Sub putball(f%) 'ボール表示・消去
    Dim c!
    If f = 0 Then c = 0 Else c = RGB(255, 255,
255)
    Picture1.Circle (blx, bly), blr, c
End Sub

Sub moveball() 'ボール移動
    Dim nblx!, nbly!
    Dim dx!, dy!, l!
    Dim ax!, ay!

    dx = mx - blx 'マウスポインタとの座標の差を
計算
    dy = my - bly
    l = Sqr(dx * dx + dy * dy) 'マウスポインタま
での距離を計算
    If l > 0 Then
        ax = dx / l '距離に応じて加速度を決定
        ay = dy / l
    Else
        ax = 0
        ay = 0
    End If
    blvx = blvx + ax * dt * g '加速度に応じて速
度を更新
    blvy = blvy + ay * dt * g
    nblx = blx + blvx * dt '速度に応じて位置を更
新
    nbly = bly + blvy * dt
    putball 0
    blx = nblx
    bly = nbly
    putball 1
End Sub

'障害物関連担当プロシージャ

Sub initbarrier() '障害物位置設定
    Dim i%
    brr = xsize / 50
    For i = 0 To 9
        brx(i) = Rnd(1) * xsize / 2 + xsize / 4
        bry(i) = Rnd(1) * ysize
    Next
End Sub

Sub putbarrier(f%) '障害物表示・消去
    Dim c!, i%
    If f = 0 Then c = 0 Else c = RGB(255, 0, 0)
    For i = 0 To 9
        Picture1.Circle (brx(i), bry(i)), brr, c
    Next
End Sub

'ゴール関連担当プロシージャ

Sub initgoal() 'ゴール位置設定
    gr = xsize / 50
    gx = xsize - gr / 2
    gy = Rnd(1) * ysize / 2 + ysize / 4
End Sub

Sub putgoal(f%) 'ゴール表示・消去
    Dim c!
    If f = 0 Then c = 0 Else c = RGB(0, 255, 0)
    Picture1.Circle (gx, gy), gr, c
End Sub
```

デジタルの世界へようこそ!

――――コンピュータの不思議をわかりやすく解説――――

第4回 逆ポーランド記法

鹿島　洋一（千葉県立八街高等学校教諭）

私たちが，ふだん使っている計算式をBASICやエクセルは，どうやって計算しているのでしょうか。たとえば

1＋（2＋3）×4

という式を計算すると答は21となりますが，コンピュータは，「括弧の中を先に計算して，足し算より先に掛け算を計算して，最後に足し算を行う」なんて考えて答を求めているのでしょうか。もっと良い方法はないのでしょうか。計算の手順をどのように決めるとコンピュータは効率よく計算してくれるのかを今月のテーマにします。

逆ポーランド記法

「a＋b」は，日本語では「aにbを加える」という言い方をします。この日本語に対応した式の書き方を「逆ポーランド記法」といいます。したがって「a＋b」は「a b＋」と書くことにします。

「1＋2×3」という式は，日本語で「1に2と3を掛けたものを加える」と言いますから，逆ポーランド記法では，「1 2 3 ×＋」と書けます。言い方を変えて「2と3を掛けたものに1を加える」と考えれば「2 3 × 1 ＋」と書くこともできます。

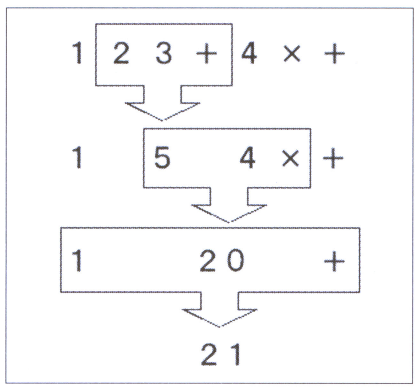
【問】逆ポーランド記法で書かれた次の式を計算しなさい

(1) 2 4 × 3 －

2と4を掛けて3を引く

(2) 3 2 ＋ 1 5＋×

3と2を足したものに1と5を足したものを掛ける



《第1図》逆ポーランド記法の求め方

もうすこし複雑な式の値を求めてみましょう。式「1＋(2＋3)×4」を逆ポーランド記法で書くと

「1 2 3 ＋ 4 ×＋」となります。

この式の，簡単に答が分かる部分を次々に計算していきましょう。「2 3 ＋」を「5」に置き換え，さらに「5 4 ×」を「20」に置き換えます。

最後に残った「1 20 ＋」を計算して，答え「21」を求めることができます(**第1図**)。

しかしこの方法で答を求めるには，最初に計算できるところを見つけなくてはいけないという欠点があります。実は逆ポーランド記法では演算の順序を意識せずに左から順に計算して答を求めることができます。この方法を知るにはスタックという考え方を理解しなければなりません。

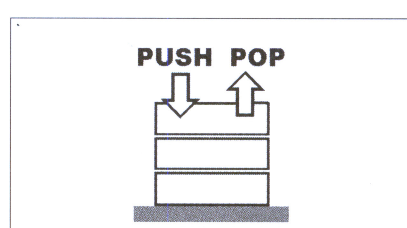
スタック

スタック[stack]とは，英和辞典で調べると「干し草の山，積み重ね」とあるように，最後に追加したデータを最初に取り出すデータの操作方法をいいます。つまり新しい商品がくる度に今ある商品の上に積み上げていく，商品は一番上のものからおろしていく，商品を入れ替えたり途中から商品を出すことはできないという「先入れ後出し」の考え方です。

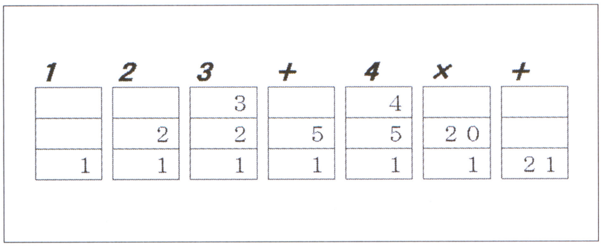
これに対して，窓口で順番を待って先着順に処理されるような「先入れ先出し」のデータ操作方法を「待ち行列」と言います。

スタックにデータを積む操作をPUSH，スタックからデータを取り出す操作をPOPといいます(**第2図**)。

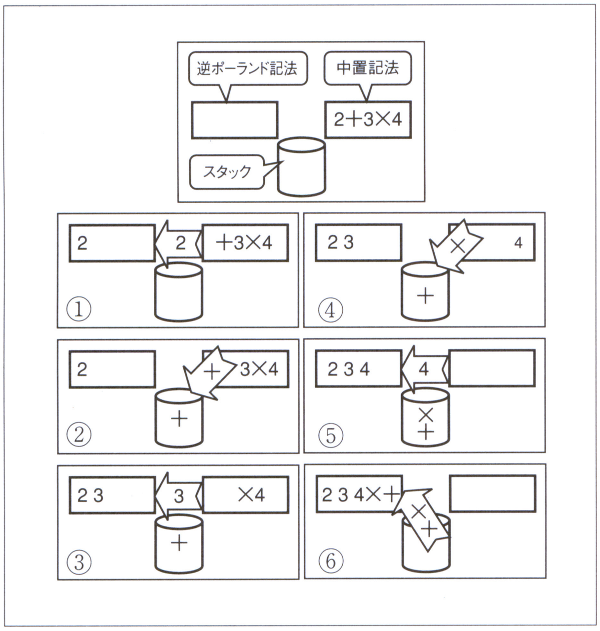
スタックはコンピュータの中ではサブルーチンなどが呼び出されたときの戻りアドレスを記録するときに使われていま



《第2図》スタッフにおけるPUSHとPOP



《第3図》スタックによる計算手順の流れ



《第4図》逆ポーランド記法に変換する手順

す。戻りアドレスをスタックに記録することで，サブルーチンからサブルーチンを呼び出すような「入れ子」になったプログラムでも正確に元の場所に戻ってくることができるのです。

式の計算

それでは，スタックの考え方を用いて「1 2 3 ＋ 4 ×＋」を計算してみましょう。計算手順は，数式を左から順に見て

(1) 被演算子（数字）なら，これをスタックにPUSHする（積む）。
(2) 演算子（＋－×÷）なら，スタックから2回POP（取り出し）を行い，計算結果をPUSHする。
(3) 式の最後に来るまで繰り返す

となります。

第3図を見ながら解説しましょう。最初のデータ「1」は数字なので，そのままスタックにPUSHします。次の「2」と「3」も同様にPUSHします。次の「＋」は演算子なので2回POPしてデータ「3」と「2」を取り出し，計算結果「5」をPUSHします。次の「4」は数字なのでそのままPUSHします。次の「×」は演算子なので2回POPして「4」と「5」を取り出し，計算結果「20」をPUSHします。最後の「＋」は演算子なので「20」と「1」をPOPして計算結果「21」をPUSHします。最後にスタックに残った数値「21」がこの式の答となります。

この記述方法は，演算子の優先順位を考える必要もないし，括弧( )の必要もない，実にコンピュータ向きのシンプルな記法というのがわかると思います。

【問題】スタックの考え方を用いて次の式を計算してみましょう。

(1) 3 4 ＋ 2 ×
(2) 1 2 ＋ 4 1 － ×
(3) 5 6 3 2 ＋ － ÷

【解】

(1) 14　(2) 9　(3) 5

式の変換

ふだん私たちが使っている式の書き方は，演算子が数字と数字の間に置かれているので，中置記法といいます。それに対して演算子を最後に置く書き方を，後置記法または逆ポーランド記法といいます。この呼び方はポーランドの論理学者が初めて使ったことから来ているそうです。

中置記法で書かれた式A×(B＋C)を逆ポーランド記法に変換してみましょう。次のような手順で変換することができます。

(1) ○と□の演算と考えます。
　　Ⓐ×(B＋C)
(2) 演算子「×」を最後に移動します
　Ⓐ (B＋C)×
(3) □の中の演算子「＋」を最後に移動します。
　Ⓐ BC＋×

これで逆ポーランド記法に変換することができました。

【問題】次の式を逆ポーランド記法に変換してみましょう。

(1) (2＋3)＋4　(2) 2＋(3＋4)
(3) (1＋2)÷3－4×(5－6)
(4) (A－B)÷(C－D)
(5) A＋B×(C－D)

【解】

(1) 2 3＋4＋　(2) 2 3 4＋＋
(3) 1 2＋3÷4 5 6－×－
(4) A B－C D－÷
(5) A B C D－×＋

スタックを用いた式の変換

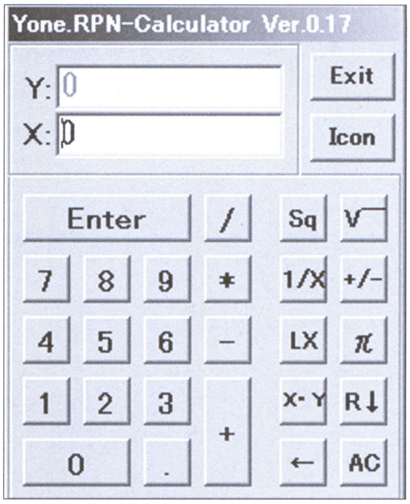
このような方法で逆ポーランド記法に変換するのは，コンピュータにとってあまり効率的な方法ではありません。これもスタックの考え方を用いて変換することで効率的に変換することができます。ここで演算子の強弱という考え方が出てきますが，これは小学校の時から知っている「足し算・引き算より，掛け算・割り算を先に計算する」というルールです。掛け算・割り算は，足し算・引き算より強く，足し算・引き算と，掛け算・割り算の強弱は同じと考えます。式で書けば

×，÷　＞　＋，－

となります。このとき変換手順は次のようになります。

(1) 数字ならそのまま書き出す。
(2) 演算子なら
　(a) スタックが空ならPUSHする。
　(b) スタックの先頭が演算子なら比べて，相手が強いか同等ならPOPして書き出す
　(c) スタックの先頭が演算子なら比べて，相手が弱ければスタックにPUSHする。
(3) 式が終わりなら，スタックのデータをすべてPOPして書き出す。

それではこの手順で　2＋3×4　を逆ポーランド記法に直してみましょう(**第4図**)。



《第5図》RPN電卓の例

①「2」は数字なので，そのまま書き出します。
②「＋」は演算子でスタックが空なので，そのままPUSHします。
③「3」は数字なので，そのまま書き出します。
④「×」は演算子なのでスタックの先頭と比べます。スタックの先頭「＋」は，「×」より弱いのでこのデータはそのままPUSHします。
⑤「4」は数字なのでそのまま書き出します。
⑥式が終わったので，スタックに残っているデータをすべてPOPして書き出します。

括弧（ ）や等号＝の入った式も，この手順で逆ポーランド記法に変換することができます。

終わりに

逆ポーランド記法(reverse Polish notation)が使える電卓をRPN電卓といいます。ヒューレット・パッカードの電卓(第5図)は，このRPNを採用しているのですが，日本HPの販売は終了してしまったようです。入手はしにくくなりましたが，この電卓の信奉者は多く，コンピュータ上でRPN電卓をシミュレートするフリーウェアもたくさんあります。そこでhttp://member.nifty.ne.jp/yshr/のフリーウェアYone.RPN-Calculatorを紹介します。RPN記法で式を書ける人は簡単に使えます。

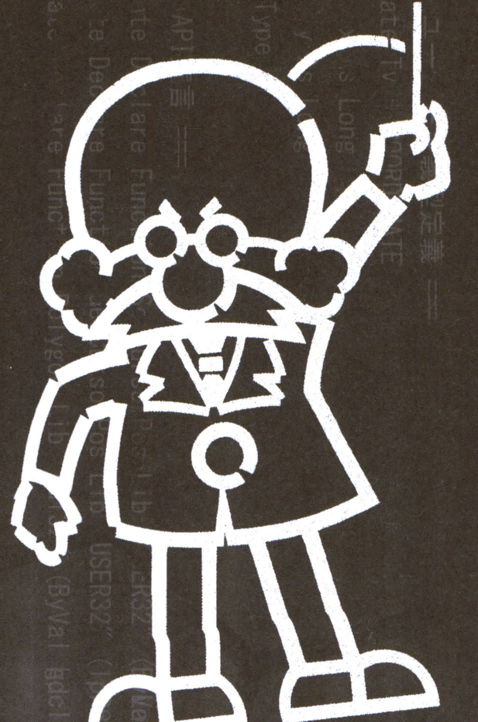
「1＋2×3」は「1 2 3×＋」となるので，　1 Enter 2 Enter 3 ＊ ＋ と入力します。詳しくは解説を読んで下さい。このソフトでRPN電卓の気分を味わってもらえれば嬉しく思います。

コラム　機械語に翻訳

私たちが，ふだん書いているBASIC等のプログラムを，コンピュータは直接理解することはできません。そこでそれをコンピュータが理解できる機械語という言語に翻訳する必要が出てきます。この仕事をするプログラムをコンパイラといいます。言語を他の言語に翻訳することはコンピュータ科学の重要な一分野となっています。

今回は「中置記法」と「後置記法」の翻訳の問題を扱いました。中置記法のままの計算が難しいのは，左から右への計算規則だけでなく，演算子の優先順位，括弧（ ）による順位の変更など，計算規則がとても複雑だからです。後置記法で表された式の計算方法を知ると，中置記法の計算がいかに複雑か理解してもらえると思います。

この二つの記法以外に「前置記法」があります。これらの記法を場面に応じて使い分けていくのがコンピュータ科学の特徴です。



プログラマー&クリエイターへの登竜門

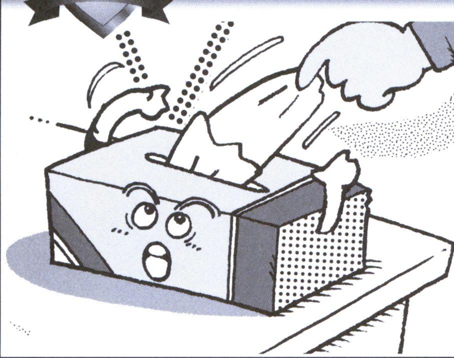
投稿プログラムコーナー BM特選

本コーナーは，さまざまな言語に対応したプログラミング学習と創作をテーマに，読者からの投稿作品で成り立っています。これから，プログラミングを勉強しようという皆さんや，プロのプログラマー/クリエイターを目指そうという皆さんにとって役立つ情報が満載です。手近にあるパソコンに入力して，お楽しみください。

それぞれの記事にはプログラムを入力/実行する時の注意点やプログラムの内容，改造方法などが紹介されていますので，よく読んでください。また，“CHECKER　FLAG”というコラムがあり，デバッグ博士ことDr.Dがそれぞれのプログラムに対し批評を加えています。口が悪い人なので，かなりキビシイことも言っていますが，自分でプログラムを作ったり，こらから投稿しようとしている人にとっては，とても参考になるでしょう。

ベスト アイデア

VB.NET



たくさん抜き取れ！

ティ取(しゅ)

松江　祐輔　（21歳）

内容とか

こんにちは。かなり久しぶりの投稿です。Visual Studio.NETを手に入れてから，何か投稿しようとは思っていたんだけど，アイデアがなかったり，時間がなかったりで，かなり経ってしまいました。

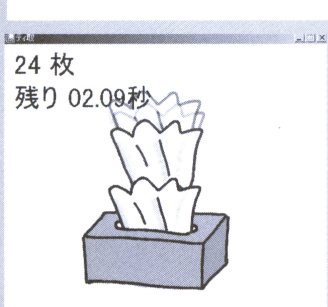
というわけで（どういうわけだ），「ティッシュペーパー取りゲー」です。10秒間でできるだけ多くのティッシュペーパーを取ってください。

ティッシュペーパーを取るには，ペーパーを上へドラッグ&ドロップしてください。ペーパーを動かすと勝手にゲームスタート。10秒経つと終了。右クリックで再ゲーム。簡単だね。

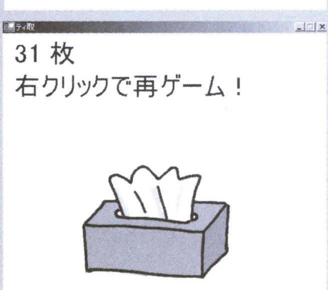
ペーパーをある程度，持ち上げて離さないと取った枚数としてカウントされません。



《写真1》ティッシュを1枚取るとゲームが始まる



《写真2》ひたすらドラッグ



《写真3》10秒経つとゲーム終了

入力方法

ホームページからファイル一式ダウンロードできるわけですが，一応，一から作るための，あまり親切でない説明です。

画像を使ってるので，テキトーでいいので画像を見ながら，tissue_box上.bmp(224×27)，tissue_box.bmp下(224×96)，tissue_paper.bmp(141×125)の3枚を描いて作ってください。透過する部分は赤色です（コードを変更すれば何色でもいいんだけど）。

まず，新しいプロジェクトを作成します。メニューの[ファイル]-[新規作成]-[プロジェクト]を選択すると，新しいプロジェクトのダイアログが表示されます。「テンプレート」は「Windowsアプリケーション」を選択し，「プロジェクト名」は「Tissue」と入力してください。

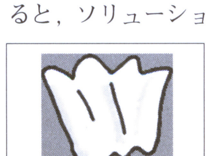
画像はリソースとして埋め込んでいるので，メニューの[ファイル]-[既存項目の追加]で作った画像を追加します。すると，ソリューションエクスプローラに画像が追加されるので選択して，プロパティの「ビルドアクション」を「埋め込まれたリソース」にしておきます。

デザイナの画面で，ツールボックスからTimerを追加します。そして，FormとTimerの**プロパティ表**を見て設定してください。FormのNameプロパティを変更しても，ファイル名はForm1.vbのままだと思うので，ソリューションエクスプローラの「Form1.vb」を選択し，プロパティウィンドウの「ファイル名」を「frmMain」に変更しておきましょう。

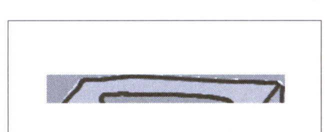
あとは，コードを打ち込むだけです。メニューの[表示]-[コード]でコードウィンドウを表示します（普通はこんなことしないで，Formをダブルクリックしたり，右クリックメニューから選んでます）。では，**プログラム・リスト**を見て打ち込んでください。そのまま全部打ち込んでも動きますが，イベント名なんかは，コードウィンドウ上部の二つのドロップダウンリストから選択して開発環境にコードを書かせましょう。例えば，左のドロップダウンリストから「(Base Class Events)」を選択して，右のドロップダウンリストから「MouseUｐ」選択すると…，ね？書いてくれたでしょ。

メニューの[デバッグ]-[開始]で実行して，きちんと動くか確認して完成です。

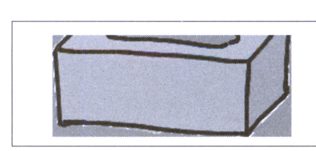
VS.NETは，ウィンドウレイアウトなど，いろいろと変更できるので，書いたとおりにウィンドウが表示されてなかっ



tissue_paper.bmp



tissue_box上.bmp



tissue_box下.bmp

《OVER FLOW》春といえば“出会い”と“別れ”。突然ですが，読者のみなさんに親しまれてきた「マイコンBASICマガジン」は，今月号をもって，休刊することになりました。永い間，本当にありがとうございました。では，最後のOFコーナーをお楽しみください。

第1表　プロパティ表

フォームの設定	
Name	frmMain
FormBorderStyle	FixedSingle
MaximizeBox	False
Size	500, 450
Text	ティ取
タイマーの設定	
Name	tmrMain
Enabled	True

たりした場合は，自分でなんとかしてくださいね。

改造のアイデア

ティッシュボックスを増やして順番に取っていくようにするとか，絵を変えて，もぐら抜きとか。ハイスコアを記録の機能を追加とか。ファイル操作は，今までのVBとだいぶ変わってます。あと，これは自分のアイデアじゃないんだけど，速く取りすぎるとティッシュペーパーが，やぶれてしまうとか。よくない？

最後に

できるだけきれいな，また.NET的なソースコードを書けるように日々頑張ってたりするわけですが，今回のコードは，ちょっとモジュールレベルの変数が多かったかなぁ。あと，OutPaperクラスは別ファイルにするべきなのかも。リストの長さとかを考えて一緒にしてみました。

それでは，ご感想などがあれば気軽に言ってくださいね。ホームページ「塊 -SOUL-」までよろしく～。http://www.alpha-net.ne.jp/users2/jz5ver1/

次はC#で投稿しようかな…。

CHECKER FLAG

影：おおっ！ついにVB.NETで投稿作品が送られてきたぜ！しかも，なかなか燃えるぜ！

編：このゲームはアイデアがいいですね。ベーマガらしいゲームに仕上がってます。

Dr.D：リストもきれいにまとめてあり，コメントが適度に入れておるので，その点も評価できるゾ！　これからVB.NETをはじめようとしている人にとってもよい作品になるな。

《リスト》ティ取のプログラム・リスト

```
Public Class frmMain
    Inherits System.Windows.Forms.Form

    Private BmpBoxSup As Bitmap
    Private BmpBoxSub As Bitmap
    Private BmpPaper As Bitmap

    Private OutCount As Integer = 0 '取ったペーパー枚数
    Private OutPapers As New Collections.ArrayList() 'ペーパーオブジェクト格納用

    '取り出したペーパーの情報を扱うクラス
    Private Class OutPaper

        '描画用
        Private matrixItems As Single()() = { _
            New Single() {1, 0, 0, 0, 0}, _
            New Single() {0, 1, 0, 0, 0}, _
            New Single() {0, 0, 1, 0, 0}, _
            New Single() {0, 0, 0, 1, 0}, _
            New Single() {0, 0, 0, 0, 1}}
        Private colorMatrix As New System.Drawing.Imaging.ColorMatrix(matrixItems)
        Private imageAtt As New System.Drawing.Imaging.ImageAttributes()
        Private alpha As Single     'アルファ値

        Public Speed As Integer      '移動スピード
        Public DirectionX As Integer '消えていくX方向
        Public Position As Point

        '描画する時必要なので ImageAttributesを外に公開
        Public ReadOnly Property ImageAttributes() As
System.Drawing.Imaging.ImageAttributes
            Get
                Return imageAtt
            End Get
        End Property

        'コンストラクタ
        Public Sub New(ByVal Position As Point)
            Me.Position = Position
            alpha = 1.0F
            imageAtt.SetColorMatrix( _
                colorMatrix, _
                Drawing.Imaging.ColorMatrixFlag.Default, _
                Drawing.Imaging.ColorAdjustType.Bitmap)
        End Sub

        'アルファ値を減らす
        Public Sub DecreaseAlpha()
            alpha -= 0.1F
            If alpha < 0 Then alpha = 0
            colorMatrix.Matrix33 = alpha
            imageAtt.SetColorMatrix( _
                colorMatrix, _
                Drawing.Imaging.ColorMatrixFlag.Default, _
                Drawing.Imaging.ColorAdjustType.Bitmap)
        End Sub

        'ペーパーが在るかどうか（アルファ値が0より大きいかどうか）
        Public ReadOnly Property IsExsit() As Boolean
            Get
                Return Convert.ToBoolean(alpha > 0.0F)
            End Get
        End Property
```

★「BASICマガジンを読む会」あらため「BASICマガジンを買ってから読む会」に改名しました。（神奈川県小田原市・**ハセ14歳**）……
【影：よしよし，それでこそ私の舎弟だ！と言いつつ，今月号で休刊なんです。ごめんなさい。また，どこかでお会いしましょう。**】**

```
End Class

Private HasPaper As Boolean = False 'ペーパーをつかんでいるかどうか
Private BoxPosition As New Point(120, 250) '箱表示位置
Private PaperPosition As New Point(120 + 40, 250 - 50) 'ペーパーの表示位置

'ペーパーをつかめる範囲
Private SeizableRect As New Rectangle( _
    BoxPosition.X + 50, _
    BoxPosition.Y - 40, _
    120, _
     70)

Private MouseCurrentPosition As Point 'マウスの現在位置
Private MouseDownPosition As Point 'ペーパーをつかんだ位置
Private SiezeTime As DateTime 'ペーパーをつかんでいた時間
Private StartTime As DateTime  'ゲーム開始時刻

'ゲームの状態
Private Enum GameState
    Ready = 0
    Playing = 1
    Finish = 2
End Enum
Private State As GameState = GameState.Ready

#Region " Windows フォーム デザイナで生成されたコード "

    Public Sub New()
        MyBase.New()

        ' この呼び出しは Windows フォーム デザイナで必要です。
        InitializeComponent()

        ' InitializeComponent() 呼び出しの後に初期化を追加します。

    End Sub

    ' Form は dispose をオーバーライドしてコンポーネント一覧を消去します。
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    ' Windows フォーム デザイナで必要です。
    Private components As System.ComponentModel.IContainer

    ' メモ : 以下のプロシージャは、Windows フォーム デザイナで必要です。
    ' Windows フォーム デザイナを使って変更してください。
    ' コード エディタは使用しないでください。
    Friend WithEvents tmrMain As System.Windows.Forms.Timer
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Me.components = New System.ComponentModel.Container()
        Me.tmrMain = New System.Windows.Forms.Timer(Me.components)
        '
        'tmrMain
        '
        Me.tmrMain.Enabled = True
        Me.tmrMain.Interval = 20
        '
        'frmMain
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 12)
        Me.ClientSize = New System.Drawing.Size(494, 418)
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.FixedSingle
        Me.MaximizeBox = False
        Me.Name = "frmMain"
        Me.Text = "ティ取"

    End Sub

#End Region

    '***************************************************************************
    'OnPaint 描画処理
    '***************************************************************************
    Protected Overrides Sub OnPaint(ByVal e As System.Windows.Forms.PaintEventArgs)
        Dim g As Graphics = e.Graphics

        '箱 上部分描画
        g.DrawImage(BmpBoxSup, BoxPosition)

        '取り出したペーパーがだんだん消えていくの処理
        If OutPapers.Count > 0 Then

            Dim i As Integer
            For i = 0 To OutPapers.Count - 1
                Dim paper As OutPaper = CType(OutPapers.Item(i), OutPaper)

                Dim x As Integer = PaperPosition.X + paper.Position.X
                Dim y As Integer = PaperPosition.Y + paper.Position.Y
                Dim w As Integer = BmpPaper.Width
                Dim h As Integer = BmpPaper.Height

                g.DrawImage( _
                    BmpPaper, _
                    New Rectangle(x, y, w, h), _
                    0, _
                    0, _
```

★ベーマガCMパロディフレーズ「まじめ，まじめ，まじめ，ベーマガ」（ってコ○トのCMそのままじゃん）。（香川県高松市・**スティック派18歳**）……【**影**：さて，そろそろベーマガもテレビコマーシャル出しますか。影主演のドラマ仕立てはどう？**編**：酒宴？】

```
                w, _
                h, _
                GraphicsUnit.Pixel, _
                paper.ImageAttributes)
        Next
    End If

    'ペーパー 描画
    g.DrawImage(BmpPaper, PaperPosition.X, PaperPosition.Y)

    'つかんでいるペーパー描画
    If HasPaper Then
        Dim x As Integer = MouseCurrentPosition.X - MouseDownPosition.X
        Dim y As Integer = MouseCurrentPosition.Y - MouseDownPosition.Y

        'ペーパーを上に持ち上げている場合のみ描画する
        If y < 0 Then
            '位置修正
            If x < -5 Then x = -5
            If x > 5 Then x = 5
            g.DrawImage(BmpPaper, PaperPosition.X + x, PaperPosition.Y + y)
        End If
    End If

    '箱 下部分 描画
    g.DrawImage(BmpBoxSub, BoxPosition.X, BoxPosition.Y + BmpBoxSup.Height)

    'ペーパーつかむ範囲 debug用
    'g.DrawRectangle(New Pen(Color.Red), SeizableRect)

    '--- 文字描画
    Dim drawFont As New Font("MS UI Gothic", 30)
    Dim drawBrush As New SolidBrush(Color.Black)

    '枚数
    g.DrawString(OutCount.ToString & " 枚", drawFont, drawBrush, 10, 10)

    '時間
    Select Case State
        Case GameState.Ready
            g.DrawString("残り 10.00秒", drawFont, drawBrush, 10, 60)

        Case GameState.Playing

            '10秒経過したら、ゲームの状態を「Finish」へ
            Dim limit As Double = 10 - Now.Subtract(StartTime).TotalSeconds
            If limit <= 0 Then
                limit = 0
                State = GameState.Finish
            End If
```

```
                g.DrawString("残り " & Format(limit, "00.00") & "秒", drawFont,
drawBrush, 10, 60)

            Case GameState.Finish
                g.DrawString("右クリックで再ゲーム！", drawFont, drawBrush, 10, 60)
        End Select

    End Sub

    '**************************************************************************
    'タイマー
    '**************************************************************************
    Private Sub tmrMain_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles tmrMain.Tick
        '取り出したペーパーの処理
        Dim i As Integer
        For i = OutPapers.Count - 1 To 0 Step -1
            'コレクションからペーパーオブジェクトを取り出す
            Dim paper As OutPaper = CType(OutPapers.Item(i), OutPaper)

            paper.DecreaseAlpha() 'アルファ値減らす

            '位置移動
            paper.Position.Y -= paper.Speed
            paper.Position.X -= paper.DirectionX

            'アルファ値が0になって見えなくなったものはコレクションから削除
            If Not paper.IsExsit Then
                OutPapers.RemoveAt(i)
            End If
        Next

        Invalidate()
        Update()
    End Sub

    '**************************************************************************
    'OnLoad
    '**************************************************************************
    Protected Overrides Sub OnLoad(ByVal e As System.EventArgs)
        Me.BackColor = Color.White

        '描画のちらつき防止用
        setstyle(ControlStyles.AllPaintingInWmPaint, True)
        setstyle(ControlStyles.UserPaint, True)
        setstyle(ControlStyles.DoubleBuffer, True)

        'リソースからBMP読み込み
        BmpBoxSub = New
Bitmap(GetType(frmMain).Assembly.GetManifestResourceStream("Tissue.tissue_box
下.bmp"))
```

★わからない！動かせない！プログラムが組めない！だったらベーマガを読め。（神奈川県相模原市・**田村なおき16歳**）……【**編**：そのとおり！バックナンバーそろってない人は，いますぐ注文だ！**影**：1997年（2月号はなし）以降ならすべて在庫があるぞ！】

```
        BmpBoxSup = New
Bitmap(GetType(frmMain).Assembly.GetManifestResourceStream("Tissue.tissue_box上.bmp"))
        BmpPaper = New
Bitmap(GetType(frmMain).Assembly.GetManifestResourceStream("Tissue.tissue_paper.bmp"))

        'BMPの透過色 指定
        BmpBoxSup.MakeTransparent(Color.Red)
        BmpBoxSub.MakeTransparent(Color.Red)
        BmpPaper.MakeTransparent(Color.Red)
    End Sub

    '*************************************************************************
    'マウスボタン アップ
    '*************************************************************************
    Private Sub frmMain_MouseUp(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles MyBase.MouseUp
        'ゲームの状態が「Finsh」の時、右クリックで「Ready」にする
        If State = GameState.Finish AndAlso e.Button = MouseButtons.Right Then
            State = GameState.Ready
            OutCount = 0 '値 初期化
            Exit Sub
        End If

        '左クリックでない または、ペーパーをつかんでいなければ処理を抜ける
        If e.Button <> MouseButtons.Left OrElse Not HasPaper Then Exit Sub

        HasPaper = False

        '紙を上へ30ドット以上取り出したら 取り出したと判断
        If MouseDownPosition.Y - e.Y > 30 Then
            'ペーパーオブジェクト生成
            Dim paper As New OutPaper(New Point(0, e.Y - MouseDownPosition.Y))

            'ペーパーが飛んで行く方向（横軸）決定
            Dim x As Integer = MouseDownPosition.X - e.X
            If x < -5 Then x = -5
            If x > 5 Then x = 5
            paper.DirectionX = x

            'つかんでから離すまでの時間でペーパーが飛ぶ速さを決定
            paper.Speed = Convert.ToInt32(((MouseDownPosition.Y - e.Y) /
Now.Subtract(SiezeTime).TotalMilliseconds) * 10 + 1)

            'コレクションに追加
            OutPapers.Add(paper)

            '取り出した枚数カウント
            If State = GameState.Playing Then OutCount += 1
        End If
    End Sub

    '*************************************************************************
    'マウス移動
    '*************************************************************************
    Private Sub frmMain_MouseMove(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles MyBase.MouseMove
        'ペーパーをつかんでいる場合
        If HasPaper Then
            MouseCurrentPosition = New Point(e.X, e.Y)

            Invalidate()
            Update()
            Exit Sub
        End If

        'マウス位置がペーパーをつかめる範囲にある場合
        'マウスポインタ形状を手の形に変える
        If SeizableRect.Contains(e.X, e.Y) AndAlso State <> GameState.Finish Then
            Me.Cursor = System.Windows.Forms.Cursors.Hand
        Else
            Me.Cursor = System.Windows.Forms.Cursors.Default
        End If
    End Sub

    '*************************************************************************
    'マウスボタン ダウン
    '*************************************************************************
    Private Sub frmMain_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles MyBase.MouseDown
        '左クリックでない または、ゲーム状態が「Finish」ならば処理を抜ける
        If e.Button <> MouseButtons.Left OrElse State = GameState.Finish Then Exit Sub

        'マウスポインタがペーパーをつかめる範囲内なら
        If SeizableRect.Contains(e.X, e.Y) Then
            HasPaper = True
            SiezeTime = Now

            MouseDownPosition = New Point(e.X, e.Y) 'クリックした位置を記憶しておく
            MouseCurrentPosition = New Point(e.X, e.Y) '現在のマウス位置

            Me.Cursor = System.Windows.Forms.Cursors.Hand

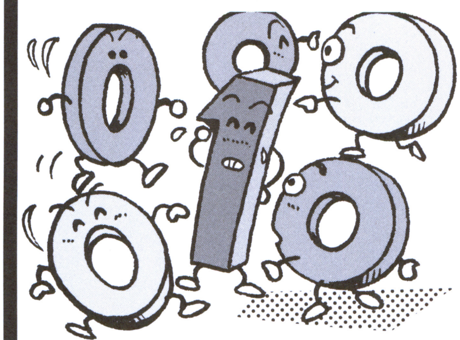
            'ゲームの状態が「Ready」なら、「Playing」状態へ
            If State = GameState.Ready Then
                State = GameState.Playing
                StartTime = Now
            End If
        End If
    End Sub

End Class
```

★2003年からの表紙に掲載されている投稿プログラム写真が面白いです。3Dっぽさを意識しているのかな？（埼玉県さいたま市・高橋博之15歳）……**【編：**4月号では，その3Dっぽさが途切れてしまいましたが，今月号は？**影：**どれどれ，あれ？**編：**どうしました？**】**

Visual Basic V6.0用

自分のレベルに合わせて楽しめる Last is one（ラスト イズ ワン）



井上　秀夫（27歳）

内容

数字をクリックして，枠内の数を減らしていくゲームです。クリックしたマスの数字はひとつ増し，同時にその上下左右のマスの数字はひとつ減ります。

最終的に，枠内に1をひとつだけ残した状態にさせることが目的です。

実行と操作

プログラムを実行後，タイトル画面(**写真1**)で，枠の大きさ，レベル，数字のフォントなどを決めます。

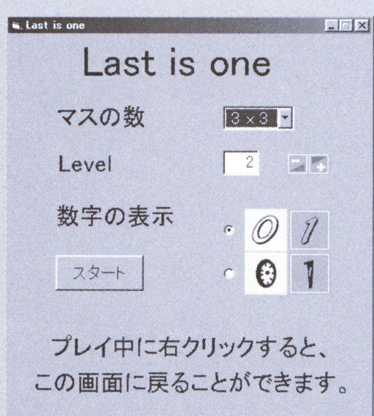
レベルの設定は，画面右の二つのボタンで調整します。押し続けると高速で加減できます。また，テキストボックスから直接入力することもできます。

各設定を行った後，スタートボタンを押すとゲームが始まります。プレイ中に右クリックすると，リプレイや設定画面へ戻ることが可能です。

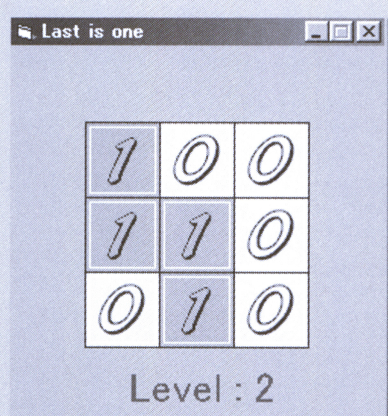
最初は感覚をつかむために，大きめの枠でレベル1から始めてみてください。プレイしてみるとわかると思いますが，レベルの数だけ手数がかかります。

レベルが上がってくると，だんだんパズルっぽくなってくるのですが，コツもいくつかあるのです。それはプレイしながら見つけてください。

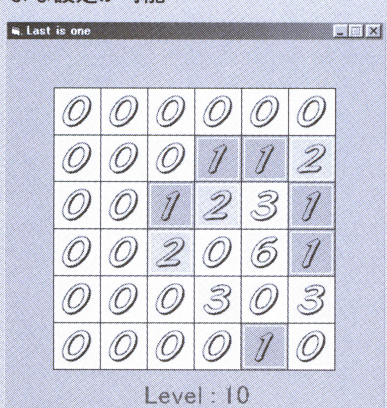
面は，手順を間違えなければ，必ず解けるようになっています。



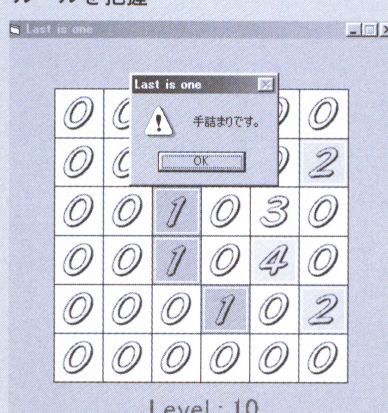
《写真1》タイトル画面で，さまざまな設定が可能



《写真2》まずは簡単なステージでルールを把握



《写真3》慣れてきたら，ハイレベルな問題にチャレンジ



《写真4》手詰まり判定もあるので，快適にプレイできる

プログラムの入力方法

フォームの配置図を参考に，ピクチャボックスを二つ，オプションボタンを二つ，コンボボックスを一つ，テキストボックスを一つ，コマンドボタンを一つ貼り付けてください。

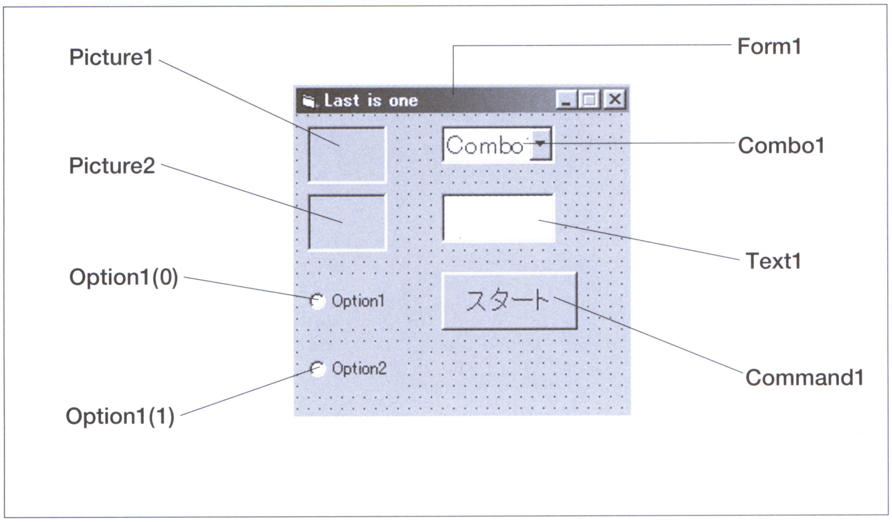
オプションボタンを二つ貼り付けた後，二つ目のオプションボタンのオブジェクト名を，一つ目と同じく「Option1」とします。すると，「コントロール配列にしますか？」と聞いてくるので，「はい」を選択します。

あとはwavファイル「clear.wav」とBMPファイル「chara1.bmp」,「chara2.bmp」を作成し，フォームとモジュールの**リスト**を入力します。

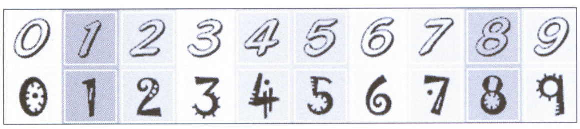
改造アドバイス

枠の大きさを増やす場合は，Form_

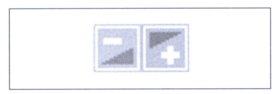
★元ベイスタでロッテに加入したローズが退団！というわけでペンネームも変更です。あ～残念無念…。（茨城県土浦市・**ローズ退団ロッテ19歳**）……【**編：**シーズンが始まる前に退団なんて，なに考えてるんでしょうね。**影：**私も退団です。またお目にかかりましょう。】



《第1図》フォームの配置図



Chara1.bmpの画像



Chara2.bmpの画像

《第2表》変数表

m(), n()	仮想画面
cx, cy	クリックした座標
px, py	数字を表示する座標
rx, ry	面作成時，数字を配置する座標
c	表示する数字
f	面セレクト時，増減の符号
h	画面上の数字の合計
ch, cha	数字のフォントの種類
lv,level	レベル
ok	手詰まり判定で使用
hen	枠の一辺のマス数
sip	面作成時，配置に失敗した回数
dame	面作成時，配置に失敗したかどうか
mode	0→プレイ中，1→設定画面，2→面作成
wmax	設定可能な枠の最大

Load内のwmaxの値を変えてみてください(例えば,8 x 8のときは，wmax=8とします)。

そしてリスト冒頭の仮想画面用の変数m(,)とn(,)内の数字は，wmaxに1を加えた値にしてください。

その他，残り手数や時間の表示などを加えてみるのも面白いと思います。

面作成の仕組み

実際のプレイと逆の動作で行っています。クリア直前の状態からランダムで適当な位置を決めて，そこから数字の加減をしています。もちろん，これもプレイの時とは，プラスとマイナスが逆です。

制作後記

最初はショートプログラムのつもりで作り始めたのですが，タイトルの設定画面のおかげ(？)で少々長くなってしまいました。

個人的には，狭い枠での高めレベルのゲームに，はまっています。

CHECKER FLAG

影：ゲームアイデア自体は目新しいものではないけど，レベルやステージの大きさなど,さまざまな設定ができるので，ゲームとしての完成度は高い作品だ。

Dr.D：プロパティをリスト内で行っている点を除いても，やはりリストがチト長めなのが残念じゃったな。じゃが，コメント記述が丁寧になれている点は評価できるゾ。

編：個人的には，BMPファイルがなくても遊べるバージョンがあると，よりベーマガ向きだったかな，と思いました。

つぐ美：井上さんの投稿作品は，遊びやすいパズルゲームが多かったという印象がありますね。現在の常連さんの中でも，昔から投稿してくださった息の長い常連さんでした。

《第1表》プロパティ表

フォームの設定	
Name	Form1
Caption	Last is one
Font	ＭＳ Ｐゴシック
…Size	14
Height	3630
Left	0
LinkTopic	Form1
MaxButton	0 'False
ScaleHeight	215
ScaleMode	3 'ﾋﾟｸｾﾙ
ScaleWidth	238
StartUpPosition	2 '画面の中央
Top	0
Width	3690
ピクチャボックスの設定	
Name	Picture1
Font	ＭＳ Ｐゴシック
…Size	9
Height	615
Left	120
ScaleHeight	37
ScaleMode	3 'ﾋﾟｸｾﾙ
ScaleWidth	53
TabIndex	0
Top	120
Width	855
Name	Picture2
Height	615
Left	120
ScaleHeight	555
ScaleWidth	795
TabIndex	6
Top	840
Width	855
コマンドボタンの設定	
Name	Command1
Caption	スタート
Enabled	0 'False
Height	615
Left	1560
TabIndex	5
Top	1680
Visible	0 'False
Width	1455
オプションボタンの設定	
Name	Option1(0)
Caption	Option1
Font	ＭＳ Ｐゴシック
…Size	9
Height	375
Index	0
Left	120
TabIndex	3
Top	1800
Width	975
Name	Option1(1)
Caption	Option2
Font	ＭＳ Ｐゴシック
…Size	9
Height	375
Index	1
Left	120
TabIndex	4
Top	2520
Width	975
テキストボックスの設定	
Name	Text1
Alignment	1 '右揃え
Height	525
Left	1560
TabIndex	2
Top	840
Width	1215
コンボボックスの設定	
Name	Combo1
Height	405
Left	1560
Style	2 'ﾄﾞﾛｯﾌﾟﾀﾞｳﾝ ﾘｽﾄ
TabIndex	1
Top	120
Width	1215

★3月号「自作一代記」にジョイスティック「XE1 Pro」が！かぁ〜懐かしい。これで遊んだゲームの数々が思い出される。(栃木県宇都宮市・**加藤従業員26歳**)……【**影：**源平，ドラスピとX68000用ソフトのデバッグにも貢献したなー。**なにわ：**遊んでただけなような。】

《リスト1》フォームのプログラム・リスト

```
Dim m(7, 7), n(7, 7), cx, cy, px, py, rx, ry
Dim c, f, h, ch, lv, ok, cha, hen, sip, dame, mode, wmax, level

'―――――――――――――――――――――――――【フォームのロード】
Private Sub Form_Load()

With Picture1
    .AutoRedraw = True
    .Width = 480
    .Height = 96
    .BorderStyle = 0
    .Visible = False
    .Picture = LoadPicture("chara1.bmp")
End With

With Picture2
    .AutoRedraw = True
    .Width = 48
    .Height = 24
    .BorderStyle = 0
    .Visible = False
    .Picture = LoadPicture("chara2.bmp")
End With

Option1(0).Width = 15:     Option1(1).Width = 15
Option1(0).Left = 245:     Option1(1).Left = 245
Option1(0).Top = 209:      Option1(1).Top = 258

Text1.Left = 245:          Command1.Left = 55
Text1.Top = 131:           Command1.Top = 250
Text1.Width = 41:          Command1.Width = 100
Text1.Height = 25:         Command1.Height = 40

Combo1.Left = 245
Combo1.Top = 81

wmax = 6
'▼ コンボボックスの内容
For i = 3 To wmax: Combo1.AddItem Str$(i) + " x" + Str$(i): Next

level = 1: settei
End Sub

'―――――――――――――――――――――――――【ゲームの設定】
Sub settei()
mode = 1
With Form1
    .AutoRedraw = True
    .Width = 6308
    .Height = 6983
    .Visible = True
    .Enabled = True
    .Cls
End With

Text1.Visible = True:          Text1.Enabled = True
Combo1.Visible = True:         Combo1.Enabled = True
Option1(0).Visible = True:     Option1(0).Enabled = True
Option1(1).Visible = True:     Option1(1).Enabled = True
Option1(ch) = True:            Text1.Text = level
startbutton

'▼ セレクトボタンを表示
BitBlt Form1.hDC, 293 + 25, 134, 48, 24, Picture2.hDC, 0, 0, &HCC0020

For i = 0 To 1 '▼ 数字のサンプルを表示
  c = i: cha = 0: px = i + 5.5
  For py = 4 To 5: hyouji: cha = cha + 1: Next py
Next i: cha = ch

Form1.Font = "westwood LET" '▼ 文字の表示
moji 83, 7, "Last is one", 35, 0
Form1.Font = "ＭＳ Ｐゴシック"
moji 55, 80, "マスの数", 20, 0
moji 57, 132, "Level", 20, 0
moji 55, 192, "数字の表示", 20, 0
moji 50, 340, "プレイ中に右クリックすると、", 20, 0
moji 30, 380, "この画面に戻ることができます。", 20, 0

End Sub

'―――――――――――――――――――――――――【面の作成】
Sub stage()
```

★今年のセンター試験BASIC，けっこう簡単でしたね。解説と同じく「距離の2乗」は，チョット無理のある表現だと思った。(神奈川県川崎市・.Local21歳) ……【編：今年も，センター試験のBASIC問題で20点もらえたという数多くのお便りが届きました。はい。】

```
'【面作成の準備】
mode = 2
Command1.Enabled = False: Command1.Visible = False
With Form1
    .Cls
    .Width = 735 * hen + 1598                    '▼ フォームの大きさ
    .Height = 735 * hen + 1883
    .Left = (Screen.Width - Form1.Width) ¥ 2      '▼ フォームの位置
    .Top = (Screen.Height - Form1.Height) ¥ 2
End With

For i = 0 To hen + 1: For j = 0 To hen + 1       '▼ 仮想画面を初期化
  m(j, i) = 0: n(j, i) = 0
  If i * j = 0 Or i > hen Or j > hen Then m(j, i) = 99: n(j, i) = 99 '▼ 範囲外は99に
Next j, i

sip = 0: lv = 0

'【面作成】
For i = 1 To level  '▼ レベルの数だけ次の処理を繰り返す
  '▼ 数字の配置位置(中央)
  Randomize: rx = Int(Rnd * hen + 1): ry = Int(Rnd * hen + 1)
  If i = 1 Then m(rx, ry) = 1  '▼ 1回目に限り、最初から中央を1に

  dame = 0 '▼ これから指定位置に配置できるか
  For j = -1 To 1 Step 2                     ' を確認する。
      If m(rx + j, ry) = 9 Or m(rx, ry + j) = 9 Then dame = 1 '▼ 周囲に9があるか確認
  Next j
  If m(rx, ry) = 0 Then dame = 1  '▼ 配置された位置が0かどうか確認
  If dame = 1 Then i = i - 1: sip = sip + 1  '▼ 配置できそうにないならやり直し
  If sip > 10 Then moji (49 * hen - 108) ¥ 2, 12, "しばらくお待ち下さい。", 18, 0

  If dame = 0 Then  '▼ 配置できるなら次の処理へ
    For j = -1 To 1 Step 2  '▼ 周囲の数をプラス1
      m(rx + j, ry) = m(rx + j, ry) + 1
      m(rx, ry + j) = m(rx, ry + j) + 1
    Next j
    m(rx, ry) = m(rx, ry) - 1  '▼ 中央の数をマイナス1
    draw  '▼ 面を描く
  End If
  DoEvents

  If sip = 1000 Then  '▼ やり直しが1000回に達したら
    Form1.Cls: mode = 0: draw: lv = level: level = i: wait 300
    MsgBox "設定レベルが高かったので、今回は Level" + Str$(i) + " でお楽しみ下さい。
", 48, "Last is one"
    i = lv
  End If
Next i

'【面作成終了】
Form1.Cls: mode = 0: draw
'▼ リプレイに備え、仮想画面に書き込み
For i = 1 To hen: For j = 1 To hen: n(j, i) = m(j, i): Next j, i
If sip < 1000 Then MsgBox "Level" + Str$(level) + "  Ready?", 32, "Last is one"
moji (49 * hen - 5) ¥ 2 - (level < 10) * 7, hen * 49 + 62, "Level :" + Str$(level),
20, vbBlue
End Sub

'――――――――――――――――――――――――――――――――【面の描写】
Sub draw()
For py = 1 To hen: For px = 1 To hen
  c = m(px, py): hyouji: Line (py * 49, px * 49)-(py * 49 + 49, px * 49 + 49), 0, B
Next px, py
End Sub

'――――――――――――――――――――――――――――――――【クリア判定】
Sub hantei()
h = 0: For i = 1 To hen: For j = 1 To hen: h = h + m(j, i): Next j, i
If h > 1 Then Exit Sub  '▼ 画面に1が一つでないなら このプロシージャを抜ける
mode = 3
Form1.Font = "westwood LET"
For i = -1 To 1: For j = -1 To 1
moji (49 * hen - 64) ¥ 2 + j, (49 * hen + 49) ¥ 2 + i, "Clear !", 40, 0
Next j, i
moji (49 * hen - 64) ¥ 2, (49 * hen + 49) ¥ 2, "Clear !", 40, &HA045FF
z = sndPlaySound("clear.wav", 1)  '▼ 音を鳴らす
Form1.Font = "MS Pゴシック": wait 1200

If sip = 1000 Then
  MsgBox "すばらしい!", 48, "Last is one"
  Form1.Cls: settei: Do: DoEvents: Loop  '▼ すぐに面作成へ進まないように
End If

level = level + 1: stage  '▼ レベルアップして次の面へ
End Sub
```

★新たに計画されている九州新幹線。なんと宮崎は通らないんですよ!ふざけるな〜。たいがいにシ〜ガイア〜。(宮崎市・**日向夏18歳**)
……【**影**:先日"はやて"に乗った。あっという間に八戸に!寝てる暇もなかった。**編**:寝てて乗り過ごしたのでは?**影**:うっ。】

```
'————————————————————————————【手詰まり判定】
Sub hantei2()
ok = 0
For i = 1 To hen: For j = 1 To hen
  If m(j, i) < 9 And m(j - 1, i) * m(j, i - 1) * m(j + 1, i) * m(j, i + 1) > 0 Then ok
= 1
Next j, i
If ok = 1 Then Exit Sub '▼ 手詰まりでなければこのプロシージャを抜ける

MsgBox "手詰まりです。", 48, "Last is one"
For i = 1 To hen: For j = 1 To hen: m(j, i) = n(j, i): Next j, i: draw '▼ リプレイ
End Sub

'————————————————————————————【数字の表示】
Sub hyouji()
BitBlt Form1.hDC, px * 49 + 1, py * 49 + 1, 48, 48, Picture1.hDC, c * 48, cha * 48,
&HCC0020
End Sub

'———————————————————————【スタートボタンの表示条件】
Sub startbutton()
Command1.Visible = False: Command1.Enabled = False
If mode = 1 And hen > 2 And level > 0 Then '▼ 設定画面で3x3以上でレベル1以上なら
  Command1.Visible = True: Command1.Enabled = True '  スタートボタンを表示
End If
End Sub

'————————————————————————————【tentou関数の作成】
Function tentou(x, y)
For k = 1 To 3
  Line (x * 49 + k, y * 49 + k)-(x * 49 + 49 - k, y * 49 + 49 - k), vbRed, B
Next k
dame = 1
End Function

'————————————————————————————【wait関数の作成】
Function wait(ByVal w As Long)
t1 = GetTickCount: t2 = t1                    '▼ t1はwait実行の瞬間
'▼ 指定した値になるまでt2に代入して待たせる
While ((t2 - t1) < w): t2 = GetTickCount: DoEvents: Wend
End Function
```

```
'——————————————————————————————————————【文字表示関数の作成】
Function moji(ByVal x, y, w, s, t As Long)
With Form1
    .CurrentX = x  '▼ X座標
    .CurrentY = y  '▼ Y座標
    .FontSize = s  '▼ サイズ
    .ForeColor = t '▼ 色
End With
Form1.Print w      '▼ 文字列
End Function

'———————————————————————————————【マウスボタンが押された時の処理】
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As Single, y As
Single)

If Button = 1 Then '【左ボタンが押されたのなら】
  Select Case mode
    Case 0 '【プレイ中の場合】
      cx = x ¥ 49: cy = y ¥ 49      '▼ ボタンを押した座標
      If m(cx, cy) > 9 Then Exit Sub '▼ 枠外をクリックしたらこのプロシージャを抜ける
      dame = 0
      For j = -1 To 1 Step 2
        '▼ ボタンを押しても変化しない場所の時は、
        If m(cx + j, cy) = 0 Then tentou cx + j, cy
        If m(cx, cy + j) = 0 Then tentou cx, cy + j '赤く点灯してその理由を知らせる。
      Next j
      If m(cx, cy) = 9 Then tentou cx, cy
      If dame = 1 Then Exit Sub
        m(cx, cy) = m(cx, cy) + 1                  '▼ クリックした所をプラス1
        For i = -1 To 1 Step 2                     '▼ 周囲の数をマイナス1
          m(cx + i, cy) = m(cx + i, cy) - 1
          m(cx, cy + i) = m(cx, cy + i) - 1
        Next i
      draw                                         '▼ 面を描く
      hantei                                       '▼ クリア判定へ
    Case 1 '【設定画面の場合】
      If (x - 30) ¥ 48 = 6 And (y - 14) ¥ 24 = 5 Then f = Sgn(x * 2 - 683): wait 160
      While f <> 0
        level = level + f: If level < 1 Then level = 1
        Text1.Text = level: DoEvents: wait 50
      Wend
    Case 2 '【面作成中の場合】
      Exit Sub
```

★ついに0xFA号ですね（あえて，16進数で記述）。1バイトだと，まさに「OVER FLOW」一歩手前です。（神奈川県茅ヶ崎市・int型22歳）……【影：FA宣言いたします。Dr.D：誰も買い手がなかったりして。影：Dr.，それはないんとちゃう？って，やっぱり？】

```
  End Select
End If

If Button = 2 Then '【右ボタンが押されたのなら】
  Select Case mode
    Case 0    '【プレイ中の場合】
      ms = MsgBox("もう一度プレイしますか？", 35, "Last is one")
      Select Case ms
        Case 6: For i = 1 To hen: For j = 1 To hen: m(j,i) = n(j,i): Next j, i: draw
                          '▼［はい］      リプレイ
        Case 7: settei    '▼［いいえ］    設定画面へ
        Case 8: Exit Sub  '▼［キャンセル］このプロシージャを抜ける
      End Select
    Case 1, 2 '【設定画面・面作成中の場合】
      Exit Sub
  End Select
End If
End Sub

'―――――――――――――――――――――――――【マウスボタンが離された時の処理】
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As Single, y As
Single)
Select Case mode
  Case 0: draw: hantei2 '【プレイ中の場合】
  Case 1: f = 0         '【設定画面の場合】
    If (x - 30) ¥ 48 = 6 And (y - 14) ¥ 24 = 5 Then f = Sgn(x * 2 - 683)
    If f = 0 Then Exit Sub
    level = level + f: If level < 1 Then level = 1
    Text1.Text = level: wait 70: f = 0
  Case 2: Exit Sub      '【面作成中の場合】
End Select
End Sub

'―――――――――――――――――――――――――――【マウスが動かされた時の処理】
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, x As Single, y As
Single)
f = 0 '▼ レベルセレクトボタン以外の場所で、レベルの上下がないように
End Sub

'――――――――――――――――――――――――――――――――――【コンボボックス】
Private Sub Combo1_Click()
hen = Val(Right$(Combo1.Text, 2)): startbutton
End Sub
```

```
'――――――――――――――――――――――――――――――――――【オプションボタン】
Private Sub Option1_Click(Index As Integer)
cha = Index: ch = cha
End Sub

'――――――――――――――――――――――――――――――――――【テキストボックス】
Private Sub Text1_Change()
level = Val(Text1.Text): startbutton
End Sub

'――――――――――――――――――――――――――――――――――【コマンドボタン】
Private Sub Command1_Click()
Text1.Visible = False:        Text1.Enabled = False
Combo1.Visible = False:       Combo1.Enabled = False
Option1(0).Visible = False:   Option1(0).Enabled = False
Option1(1).Visible = False:   Option1(1).Enabled = False
stage
End Sub
```

《リスト２》モジュールのプログラム・リスト

```
Option Explicit

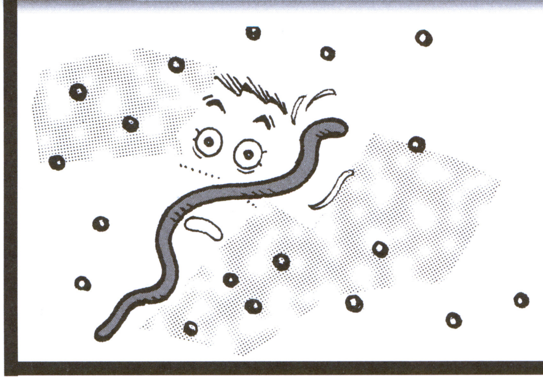
'【キャラクタ表示のため】
Declare Function BitBlt Lib "gdi32" ( _
ByVal hDestDC As Long, _
ByVal x As Long, ByVal y As Long, _
ByVal nWidth As Long, ByVal nHeight As Long, _
ByVal hSrcDC As Long, _
ByVal xSrc As Long, ByVal ySrc As Long, _
ByVal dwRop As Long) As Long

'【wait関数作成のため】
Declare Function GetTickCount Lib "kernel32" () As Long

'【音を鳴らすため】
Declare Function sndPlaySound Lib "winmm.dll" Alias "sndPlaySoundA" _
(ByVal lpszSoundName As String, ByVal uFlags As Long) As Long
```

★しばらくベーマガから遠ざかっていたのですが，250号と知って久々に購入しました。長い歴史には感服いたします。（静岡県掛川市・？？31歳）……【編：250号に関するコメントがたくさん届いています。ありがとうございます。残念ですが，今月号で終了です。】

Visual Basic V6.0用



触手を操って点をつぶす

点(てん)つぶし

正凹 (17歳)

作品の内容

いわゆるワンキーゲームと呼ばれる点避けゲームは数多く存在しますが，このゲームでは逆に点をつぶしまくります。

一気に点をつぶして高得点を狙え！

操作方法

プログラムを実行するとタイトルが表示されるので(写真1)，[Enter]キーを押してゲームをスタートしてください。

ゲーム中の操作は，スペースキーを押したり離したりするだけ！

画面の一番左端にいる，周りが緑色で中心が白い円が自機です。スペースキーを押すと自機が上昇し，離すと下降します。

自機の動きに合わせて黒い触手がついてくるので，これを赤い点にぶつけまくってください(写真2)。

一振りで赤い点をいっぱい潰すと連鎖となり，高得点を得られます(写真3)。

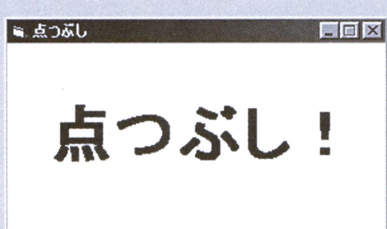
自機に赤い点が当たるか，画面上端下端から自機がはみ出してしまうとゲームオーバーです(写真4)。

点数，連鎖などの記録を保存しますので，頑張って凄い点数を叩きだしてください。

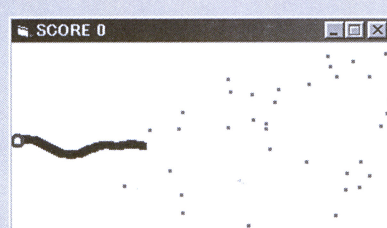
記録を初期化する場合は，本プログラムなどと同じフォルダにある，“点つぶし.dat”というファイルをごみ箱に捨ててください。

制作に苦労した点

黒い触手や赤い点を描くのに，当初は



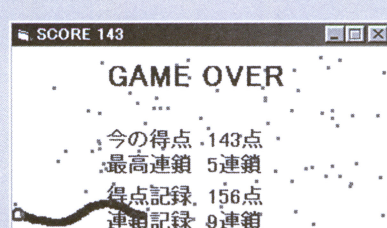
《写真1》文字どおり，点をつぶすゲーム



《写真2》迫り来る点に触手をぶつけろ！



《写真3》ただいま7連鎖中。高得点を狙おう



《写真4》自機が，1度でも点にぶつかるとゲームオーバー

《第1表》プロパティ表

フォームの配置図	
Name	Form1
BackColor	&H00C0FFFF&
BorderStyle	1 '固定(実線)
Caption	点つぶし
DrawWidth	3
FillStyle	0 '塗りつぶし
ForeColor	&H00C000C0&
Height	2625
Left	0
LinkTopic	Form1
MaxButton	0 'False
ScaleHeight	150
ScaleMode	3 'ﾋﾟｸｾﾙ
ScaleWidth	300
StartUpPosition	2 '画面の中央
Top	0
Width	4590
ピクチャボックスの設定	
Name	P1
AutoRedraw	-1 'True
BorderStyle	0 'なし
Height	135
Left	2640
ScaleHeight	9
ScaleMode	3 'ﾋﾟｸｾﾙ
ScaleWidth	6
TabIndex	0
Top	960
Visible	0 'False
Width	90

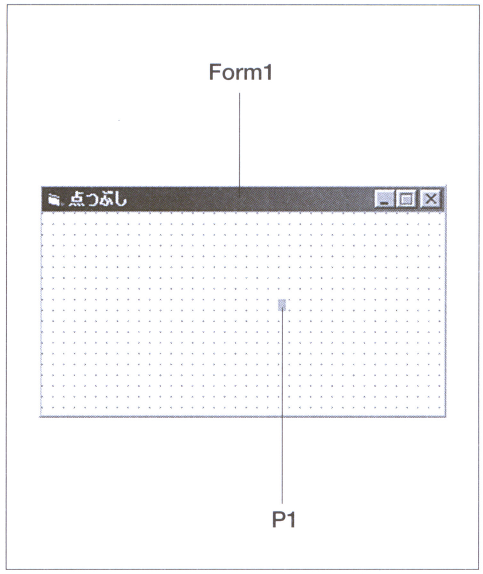
★近所のリサイクルショップで，電波新聞社製のゲームを発見しました。マッピーに似てるような…。(秋田県本荘市・**貴穂18歳**)……【**編：**電車の中で影さんに似ている人を発見。マスクにサングラスをかけ，ケイタイいじってました。彼女と一緒だったから違うか。】

LineメソッドやPsetメソッドを使っていたのですが，いろんなWebページを見てまわると「VBのメソッドなんて激遅ちゃんだよ。Win32APIを使うよね，ふ・つ・う」というような記述がいっぱいあり，突如，不安感に襲われたのでWin32APIを使うことにしました。

しかしSetPixelVでは描く点が小さすぎるんですよね。無理矢理BitBltで代用してしまいました。結果的には，思ったより速くならなかったし，PictureBoxも追加するわプログラムも長くなるわで…。

作者のホームページで，今回の作品のパワーアップバージョンを掲載しています。

http://mr-masa.hp.infoseek.co.jp/



《第1図》フォームの配置図

CHECKER FLAG

影：単純なゲーム内容だけど，見た目以上の難易度を誇るゲーム。得点や連鎖数にこだわりだすと，ついつい遊んでしまうゲームだ。

つぐ美：ステージ自体が斜めにスクロールする機能があり，それがゲームの難易度を上げていますね。

Dr.D：特別，目新しい処理はないのじゃが，スコアファイル作成，MIDI再生，触手の描画，そして丁寧なコメント記述など，基本的な要素をしっかり押さえておるリストじゃ。

編：VBメソッドとWin32APIについては，描画速度を比較する簡単なベンチマークプログラムなどを制作して，研究してみても面白いのではないでしょうか？

《リスト》点つぶしのプログラム・リスト

```
Option Explicit
'キーが押されているかどうかを調べる関数-----------------------
Private Declare Function GetAsyncKeyState Lib _
"USER32" (ByVal vkey As Long) As Integer
'Windowsが起動してから何ミリ秒たっているかを調べる関数----------
Private Declare Function timeGetTime Lib _
"winmm.dll" () As Long
'指定した座標のRGB値を取得(Pointメソッドみたいなもの？)--------
Private Declare Function GetPixel Lib "gdi32" _
(ByVal hdc As Long, ByVal X As Long, ByVal Y As Long) As Long
'音楽がいらないなら消す---------------------------------------
Private Declare Function mciSendString Lib "winmm.dll" _
Alias "mciSendStringA" _
(ByVal lpstrCommand As String, _
ByVal lpstrReturnString As String, _
ByVal uReturnLength As Long, _
ByVal hwndCallback As Long) As Long '

Dim retString As String * 256
'絵を転送する-------------------------------------------------
Private Declare Function BitBlt Lib "gdi32" _
(ByVal hDestDC As Long, ByVal X As Long, ByVal Y As _
Long, ByVal nWidth As Long, ByVal nHeight As Long, ByVal _
hSrcDC As Long, ByVal xSrc As Long, ByVal _
ySrc As Long, ByVal dwRop As Long) As Long

Const srccopy = &HCC0020 'そのままコピー
'他の変数-----------------------------------------------------------
Dim ret As Long 'API関数の戻り値を入れる
Dim TX() As Integer, TY() As Integer, TK As Integer '敵のX座標Y座標と敵の数
Dim HJ As Integer '雑用
Dim GA As Boolean 'ただいまゲームを開始しているかしていないのか
Dim C As Long 'カウンタ変数
Dim X() As Integer, Y() As Single 'プレイヤーとその触手のX座標Y座標
                                   'X(0),Y(0)…プレイヤー それ以外…触手
Dim YS As Single 'プレイヤーの上下移動速度
Dim HURU As Integer '敵の上下移動速度
Dim REY As Single '移動前の触手の末端の座標を入れる
Dim REN As Integer '連鎖回数を入れておく
Dim SC As Single, HSC As Long 'このゲームの点数,今までのハイスコア
Dim GREN As Integer, HREN As Integer 'このゲームの最高連鎖,今までの最高連鎖
Dim SetPath As String '保存するところのパスを取得

Const TekiTime = 30 '敵の移動の向きが変わる頻度。小さいほどよく変わる
Const TekiHajime = 50 '敵の最初の数
Const TekiH = 1 '敵を倒せなかった時に増える敵の数
Const KAZU = 50 '自分の触手の長さ
Const RenSinPuku = 1 '連鎖が可能になる最低限の触手の振幅。大きくするほど難しくなる
Const BgmFile = "C:¥WINDOWS¥MEDIA¥Passport.mid" '音楽ファイル
Const Wt = 60 'ウェイト調整  小さいほど速くなる
```

```
'GameOver時に表示する文字列の位置を一括してまとめたもの
Const GAMESETTEi = "18007501012057506512057508512047511012047513O"

'エンターキーを押してゲームをスタート---------------------------
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
  If KeyCode = vbKeyReturn And GA = False Then
    P1.Line (0, 0)-(3, 3), QBColor(12), BF 'ピクチャボックスに赤い点の絵を描く
    P1.Line (0, 3)-(6, 8), QBColor(0), BF  'ピクチャボックスに触手の絵を描く
    GA = True 'ゲームが始まったというフラグを立てる
    GREN = 0: SC = 0: YS = 0: HURU = 0: HJ = 0: TK = TekiHajime '変数の中身を初
期値に戻す
    ReDim TX(TK) As Integer, TY(TK) As Integer '敵（赤い点）の数を初期値に戻す
    Me.FillColor = QBColor(15)

    For C = 0 To KAZU '触手を初期位置に戻す
      X(C) = 4 + C * 2
      Y(C) = 75
    Next C

    For C = 0 To TK '敵（赤い点）の始めの出現位置を決める
      TX(C) = Int(Rnd * 300) + 300
      TY(C) = Int(Rnd * 151)
    Next C

    Me.Caption = "SCORE 0"
    Call Main
  End If
End Sub

Private Sub Form_Load()
  Dim free As Integer '空いているファイル番号を入れる
  ReDim X(KAZU) As Integer, Y(KAZU) As Single
  On Error GoTo 10
    GA = False

    SetPath = App.Path
    If Right(SetPath, 1) <> "¥" Then SetPath = SetPath & "¥"

    free = FreeFile
    Open SetPath & "点つぶし.dat" For Binary Access Read As #free 'ファイルを読
み込む
      Get #free, , HSC  'ハイスコアを読み込む
      Get #free, , HREN '今までの最高連鎖を読み込む
    Close #free
10
  Randomize
  Me.DrawWidth = 3
End Sub

Sub Main()
  ret = mciSendString("play bgm", vbNullString, 0, 0) '音楽がいらないなら消す
```

★「つまり2進数の次にはヒヤシンスーが来るであろうと！」「それ花でしょう？！」（愛知県名古屋市・**ズオウ・リアキ30歳**）……**【影：**パソコン講習会で。おばあさん「おにぎりにスイカ？」講師「ですから，お気に入りに追加です」。**編：**ま，ありがちなダジャレですね。**】**

```
  While GA = True
    Call Sousa
    Call Syori
    Call Kaku
    Call Wait(1)
    Call Bgm '音楽がいらないなら消す
  Wend

  Call GameOver
End Sub

Sub Sousa() '自機の操作
  If GetAsyncKeyState(32) <> 0 Then 'スペースキーが押されたなら
    YS = YS - 0.15 '上昇
  Else
    YS = YS + 0.15 '下降
  End If

  If YS > 4 Then YS = 4 '下降スピードの最大値の設定
  If YS < -4 Then YS = -4 '上昇スピード最大値の設定
End Sub

Sub Syori() '敵キャラの移動処理や判定
  Dim D As Long 'カウンタ変数
  Y(0) = Y(0) + YS
  REY = Y(KAZU)
  For C = KAZU - 1 To 0 Step -1
    Y(C + 1) = Y(C)
  Next C
  If Abs(REY - Y(KAZU)) < RenSinPuku Then REN = 0 '触手の振幅が規定値より小さくな
ったら連鎖を止める

  For C = 0 To TK
    TX(C) = TX(C) - 2: TY(C) = TY(C) + HURU '敵の縦横への移動
    If TX(C) < 0 Then
      TX(C) = 300 + Int(Rnd * 301): TY(C) = Int(Rnd * 151) '敵を左端→右端へ移動す
る
      TK = TK + TekiH '敵の数を増やす
      ReDim Preserve TX(TK) As Integer, TY(TK) As Integer
      For D = TK - TekiH To TK
        TY(D) = Int(Rnd * 151)
        TX(D) = 300 + Int(Rnd * 301) '新しく増えた敵の出現位置を決める
      Next D
    End If
    If TY(C) < 0 Then TY(C) = 150 '敵が上までいくと、下に移動する
    If TY(C) > 150 Then TY(C) = 0 '敵が下までいくと、上に移動する
    If GetPixel(Me.hdc, TX(C), TY(C)) = QBColor(0) Then '色で触手に敵が当たってい
るか判定
      TX(C) = 300 + Int(Rnd * 301): TY(C) = Int(Rnd * 151) '敵を右端に移動する
      REN = REN + 1: SC = SC + 1 + REN * 0.5 '点数と連鎖回数を増やす
    End If
    If Sqr((X(0)-TX(C)) ^ 2 + (Y(0)-TY(C)) ^ 2) <= 5 Then GA=False '自機と敵の当た
り判定
  Next C

  If Y(0)<0 Or Y(0)>150 Then GA=False '自機が画面からはみ出した時、GameOverのフラ
グを立てる

  HJ = HJ + 1
  If HJ >= TekiTime Then
    HURU = Int(Rnd * 5) - 2 '敵（赤い点）の上下移動速度を変える
    HJ = 0
  End If

  If REN > GREN Then GREN = REN '連鎖回数がこのゲームで最高なら、最高連鎖と置き換
える
End Sub

Sub Kaku() '表示
  Static MSC As Single, MREN As Integer
  Me.Cls

  For C = 0 To KAZU '触手を描く
    ret = BitBlt(Me.hdc, X(C) - 3, Y(C) - 3, 6, 6, P1.hdc, 0, 3, srccopy)
  Next C

  Me.Circle (X(0), Y(0)), 4, QBColor(3)

  For C = 0 To TK '敵を描く
    ret = BitBlt(Me.hdc, TX(C) - 1, TY(C) - 1, 3, 3, P1.hdc, 0, 0, srccopy)
  Next C

  If MSC <> SC Or MREN <> REN Then
    Me.Caption = "SCORE " & Int(SC) 'タイトルバーに点数の表示
    MSC = SC
    MREN = REN
    If REN > 1 Then
      Me.Caption = Me.Caption & "     " & REN & "連鎖" '連鎖回数の表示
    End If
  End If

End Sub

Sub Wait(ByVal W As Integer) 'ウェイト
  Static WW As Long 'ウェイト用の変数

  Do
      DoEvents
  Loop Until timeGetTime - WW >= Wt * W
  WW = timeGetTime
End Sub

Sub Bgm() '音楽がいらないなら消す
  ret = mciSendString("status bgm mode", retString, 255, 0)

  If Left$(retString, 7) = "stopped" Then
    ret = mciSendString("play bgm from 0", vbNullString, 0, 0)
  End If
End Sub

Private Sub Form_Paint()
   On Error Resume Next
    ret = mciSendString("open """ & BgmFile & """ alias bgm", _
    vbNullString, 0, 0) '音楽がいらないなら消す
   Me.FontSize = 38: Me.ForeColor = QBColor(0)
   Me.CurrentX = 33: Me.CurrentY = 45: Me.FontBold = True
   Me.Print "点つぶし！"
   Me.AutoRedraw = True
End Sub

Private Sub Form_Unload(Cancel As Integer)
  Dim free As Integer
  ret = mciSendString("close bgm", vbNullString, 0, 0) '音楽がいらないなら消す

  free = FreeFile
  Open SetPath & "点つぶし.dat" For Binary Access Write As #free 'ファイルに書き
込む
    Put #free, , HSC 'ハイスコアを書きこむ
    Put #free, , HREN '今までの最高連鎖
  Close #free
  End
End Sub

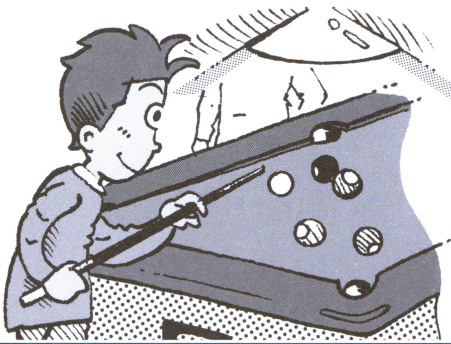
Sub GameOver()
  ret = mciSendString("stop bgm", vbNullString, 0, 0) '音楽がいらないなら消す

  SC = Int(SC) '点数の小数点は切り捨て
  If SC > HSC Then HSC = SC 'ハイスコアの更新
  If GREN > HREN Then HREN = GREN '今までの最高連鎖記録の更新

  For C = 0 To 4 'ハイスコア等の表示
      Me.FontSize = Val(Mid$(GAMESETTEi, C * 9 + 1, 2))
      Me.ForeColor = QBColor(Val(Mid$(GAMESETTEi, C * 9 + 3, 2)))
      Me.CurrentX = Val(Mid$(GAMESETTEi, C * 9 + 5, 2))
      Me.CurrentY = Val(Mid$(GAMESETTEi, C * 9 + 7, 3))
    Select Case C
      Case 0
        Me.Print "GAME OVER"
      Case 1
        Me.Print "今の得点  " & SC & "点"
      Case 2
        Me.Print "最高連鎖  " & GREN & "連鎖"
      Case 3
        Me.Print "得点記録  " & HSC & "点"
      Case 4
        Me.Print "連鎖記録  " & HREN & "連鎖"
    End Select
  Next C
  Erase TX, TY
End Sub
```


★花粉症で，鼻はでるわ，目はかゆいわで，どうにかならないかなー。みなさんは大丈夫ですか？（神奈川県厚木市・**ぴいちゃ31歳**）
……**【編：**花粉症のみなさんに！こんなファッションはいかがでしょう。モデルは影さんです。**影：**サングラスとマスクは必需品。**】**

Visual Basic V6.0用



完成された玉の動きに注目

ビリヤード

峠　恒司（48歳）

はじめに・ルール

ビリヤードのいわゆる9ボールです。マウスを使用してゲームをプレイします。ゲームのルールと操作方法は，以下のように設定されています。

・マウスでクリックした場所に向かって白玉が動きます

・白玉とクリックした場所との距離に比例して動く長さが変わります

・マウスを右クリックすると左をクリックしたときの半分しか動きません

・1から9までのボールを順番に落としていき，全部落としたらクリア。全部落とさないといけないところが，実際のルールとは違うところです。20回以内で落としたらＣＧを見る事ができます

・玉のスピードを調節できます。速すぎるときは数値を大きく(40〜50)してください

Windows98で作ったのですが，XPで動かすとオーバーフローのエラーが出ました。それも今まで作ったゲームすべて!!

何日も悪戦苦闘しましたが，結局BITBLT関数のAPIの書き方が古かったようです。わかってしまえば何のこともないのですが…。

Pick up LIST

《リスト》マウスクリック時の処理を行う関数のリスト

```
Private Sub Picture1_MouseDown(BUTTON As
Integer, Shift As Integer, X As Single, Y As
Single)

Dim J As Integer, SP As Integer
Dim J1 As Single, J2 As Single, J3 As Single, J4
As Single

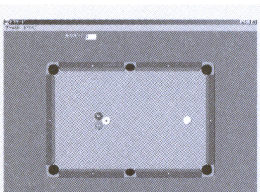
If LEV = 0 Then
    LEV = 1: Call NEWGAME
    Exit Sub
End If

If MSM > 0 Then Exit Sub
MSM = 1
If Abs(X - EX(0) - 12) - Abs(Y - EY(0) - 12) >=
0 Then
    X1(0) = Sgn(X - EX(0) - 12)
    Y1(0) = (Y - EY(0) - 12) / Abs(X - EX(0) -
12)
Else
    Y1(0) = Sgn(Y - EY(0) - 12)
    X1(0) = (X - EX(0) - 12) / Abs(Y - EY(0) -
12)
End If
J = Int(Sqr((X - EX(0) - 12) ^ 2 + (Y - EY(0) -
12) ^ 2))

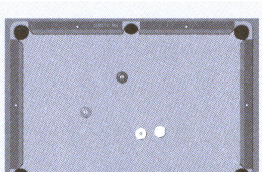
Select Case J
    Case 351 To 1000
        DIST(0) = J ¥ 5
    Case 251 To 350
        DIST(0) = J ¥ 4
    Case 150 To 250
        DIST(0) = J ¥ 3
    Case 71 To 149
        DIST(0) = J ¥ 2
    Case 30 To 70
        DIST(0) = J ¥ 1.5
    Case Else
        DIST(0) = J - 5
End Select

If BUTTON = 1 Then DIST(0) = DIST(0) + (DIST(0)
¥ 2)
Z = SNDPLAYSOUND(CurDir$ + "¥shu.wav", 1)
SP = Val(Text1.Text): If SP < 1 Then SP = 1
Timer1.Interval = SP

End Sub
```



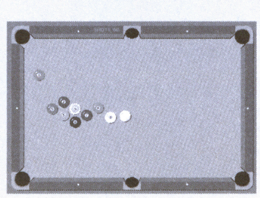
《写真1》本格派ビリヤードゲーム



《写真2》まずは，レベル1でプレイしよう



《写真3》すべての玉を落とせばクリア



《写真4》レベル3でのクリアを目指せ！

おわりに

はじめは，玉を打つ強さを決めるようにしていたのですが，ちょっと操作が面倒です。それでこのようなルールになったのですが，ストレスなくできます。

ダウンロードファイルに収録されているプログラム·リストには，あまりコメントがありませんが，難しいことはしていません。

CHECKER FLAG

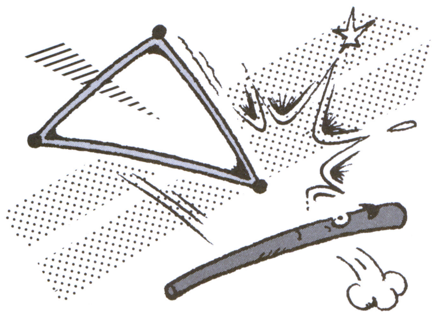
編：アイデア面での目新しさはゼロに近いですが，ゲームそのものは，ものすごくしっかりと制作されています。まさに超大作にふさわしい完成度です。

つぐ美：レベル3でのクリアは，けっこう難しいかもしれませんね。ちなみに20回以内のショットでクリアすると，ご褒美CGが見られます。

Dr.D：Pick Up LISTの補足説明をしておくと，このゲームではマウスクリックによる処理は座標設定などで，ゲーム中の玉の動きは，タイマーによる呼び出しに同期させておこなっておる。そのため，このプログラムの処理を研究したいという読者は，NINE.frmリスト内のTimer1_Timer()関数内をチェックすると良い。ただし，コメント記述がほとんどないので，リスト解読は苦労するじゃろう。

★3月号90ページのCHECKER FLAGには考えさせられました。もっとしっかりプログラムしていくため，気合をいれます。（神奈川県川崎市・色落ち18歳）……【影：気合を入れなおしてがんばるぞー。つぐ美：では，しっかり歯をくいしばってー！影：うっ！あわわ。】

HSP V2.55用



図形の頂点を弾け！

Line Ball（ライン ボール）

稲津　和磨（17歳）

使用機種・言語

遅い学校のパソコンでも動くように作ったので，最近のパソコンならまず動くでしょう。言語はおにたま氏のHSP2.5です。

内容

内容はボールをパドルで弾くゲームですが，見た目が図形の頂点を弾くようになっています。スピードが速すぎるときは*mainのウェイトを調節するか，頂点の基本移動スピードを上げて下さい。

操作方法

マウスを左右に動かすことでパドルを操作します。

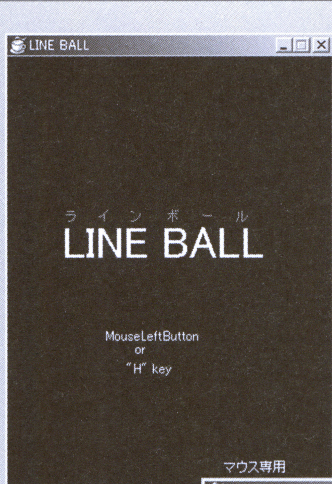
2頂点がほぼ同時に落ちてきても，なんとかなります。素早くマウスを動かしましょう。また，マウスはパドルの近くに置いておくのが良いでしょう。

画像は内部で作ります。

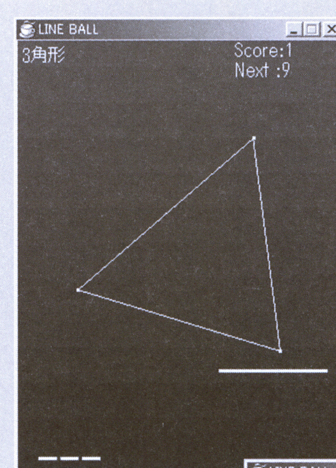
音のデータ/mo.wav/パドル跳ね返り音（ないときはsnd～を削除）

プログラムの仕組み

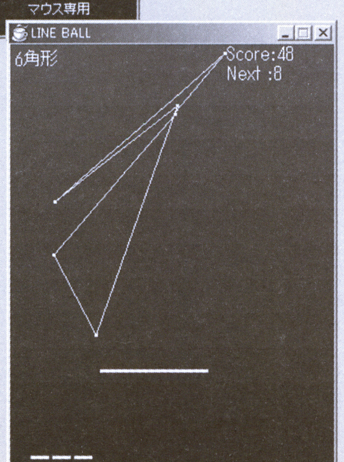
なるべくメインループがごちゃごちゃにならないように，プログラムを分けました。HSPのawaitは前のawaitからの時



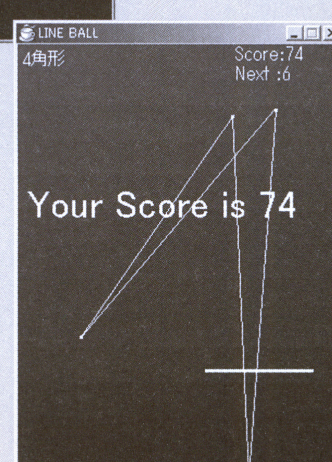
《写真1》タイトル画面。Hキーでヘルプを呼び出せる



《写真2》さあ，頂点をはじこう！



《写真3》数が増えてくると大変



《写真4》残機がなくなるとゲームオーバー

★祝！250号！はじめてベーマガを買ったのが17年前。Dr.Dはあの当時と同一人物なのですか？もしや2代目！？（東京都小平市・ゆっきー28歳）……【Dr.D：誰が2代目じゃ。ワシはワシ1人じゃ。影：頭から湯気出るほど怒らなくても。2代目に引き継ぎますか？】

間は休む仕様だったので，速度については安定しています。細かいことはプログラム中にコメントを入れ書いてあります。

改造のアドバイス

背景をかっこよくする/ハイスコア残す/BGMをつける，などなど…

苦労したところ

パレットの使用。始めは画像を別に用意していたけど，投稿ってことで何とか内部で作りたかったし，手軽なフェードインがしたかったのでパレットを使うことにしました。慣れないと難しいです。(その甲斐あって綺麗！)

構造体のようなものがなく，配列を使ったので何番のデータが何なのかが描いていて分からなくなったりしました。

コメント

このゲーム作成には，プログラム作成HSP　v2.5（作者：おにたま氏），WAVE作成　Swave(作者：Ｓｈｏｕ氏)。以上のものを使いました。どれもとても使いやすいです。

このソフトは2年ほど前に学校のパソコンで「スクリーンセーバー」を見て思いつきました。その後3回同じ物を作りましてこれが3号目です。かなり小さめにできたと思っています。

やっと投稿できます。ベーマガは毎月本当に参考になります。これからもゲームを作ります，よろしくお願いします(JAVA+Delphiの勉強中！！)。

《第１表》変数表

ball	各頂点のデータ（２次元配列）
bmax	頂点の数
re	残機
bl	パドルの長さ（定数扱い）
bx	パドルの位置（マウスと連動）
scr	スコア
nol	次の面へ行くのに必要な得点
keys	stick命令用キー情報
kk	汎用
out	落ちたかどうか？（*bmoveの戻り値みたいなもの）

ＨＰを作りました！！HSPのゲームなんかを置いています。ぜひのぞいて見てください（まだしょぼいけど）。http://www.geocities.co.jp/NeverLand/2409/)

ちなみに僕はネットにいけないので，いつも友達にUPしてもらっています。ThankYou！

CHECKER FLAG

編：こういう改造しやすい作品は，初心者にとってよいお手本になりますね。

Dr.D：確かにそうじゃ。ゲームとしてはただボールが増えていくだけなので，もう一工夫ほしいゾ！読者のみんなはぜひ改造してみてくれ。

つぐ美：でも，背景のグラデーションや，頂点を線で結ぶといった演出もセンスいいわ。この点は十分評価できますよ。

《リスト》Line Ballのプログラム・リスト

```
  width 300,400
    dim ball,50,5               ;ライン頂点の管理変数
    ;0 x
    ;1 y
    ;2 xv
    ;3 yv

    buffer 2,300,400+3,1        ;画像格納用バッファ
*bmake                          ;画像を作る
    repeat 200
    palette cnt,0,0,cnt         ;パレット作成
    loop
    palette 200,255,100,100     ;タイトルルビ
    palette 201,255,255,255     ;白
    palette 202,0,0,0           ;黒


    palfade                     ;パレット反映

    repeat 400                  ;バック作り
    palcolor 199-(cnt/2)
    line 0,cnt,300,cnt
    loop


    screen 0,300,400,1          ;メイン画面初期化

    randomize                   ;乱数初期化


    sndload "mo.wav",0,0        ;パドル跳ね返り音

*st
    bmax=3                      ;bmax-角形
    re=3                        ;残機
    bl=100                      ;バーの長さ（変動なし）
    scr=0                       ;スコア
    nol=10                      ;Nextまでの数

*ti                             ;タイトル画面
    title "LINE BALL"
    cls 4
    palcopy 2                   ;パレットを合わせる

    font "MSゴシック",12
    palcolor 200
    pos 50,140
```

★ハガキをはさみで切るの面倒です。できたらミシン目を入れてください。（東京都渋谷区・**祐天寺30歳**）……【**影**：読者はがきの切り取りネタは，ついに最後の最後まで絶えることがありませんでしたね。**編**：かたくなに，ミシン目を入れなかったベーマガでした。】

```
    mes "ラ　　イ　　ン　　ボ　　ー　　ル"
    font "MSゴシック",40
    palcolor 201
    pos 50,150
    mes "LINE BALL"

    font "MSゴシック",12
    pos 90,250:mes "MouseLeftButton"
    pos 110,262:mes "  or"
    pos 110,280:mes "¥"H¥" key"

    pos 200,370
    mes "マウス専用"
*@
    stick keys
    await 10
    if keys=256:goto *@f        ;マウス専用なのでマウスのクリックのみ反応

    getkey kk,72:if kk=1:goto *explan    ;簡易ヘルプ
    goto *@b
*@

*init                           ;各種初期化

    repeat bmax
    rnd ball.cnt.0,400          ;頂点x
    rnd ball.cnt.1,200          ;頂点y

    rnd ball.cnt.2,7            ;頂点x移動量
    ball.cnt.2-=3
    rnd ball.cnt.3,5            ;頂点y移動量
    ball.cnt.3-=2

    if ball.cnt.2=0:ball.cnt.2=1    ;真上真下真横を防ぐ
    if ball.cnt.3=0:ball.cnt.3=1
    loop
    palfade 0                   ;2回目以降のパレット初期化用
    font "MSゴシック",18
    gosub *g                    ;背景としてゲーム画面を描く
    palcolor 255
    font "MSゴシック",30
    pos 100,100
    mes "Ready?"
*@
    stick keys
    await 10
    if keys=256:goto *@f        ;マウス左
    goto *@b
*@
```

```
    font "MSゴシック",18
*main                           ;メインルーチン
    redraw 0
    gosub *g                    ;画像を描く
    redraw 1

    gosub *bmove                ;頂点を動かす
    if out=1:re--:bmax--:gosub *fadein:if re!-1:goto *init:else:goto *over    ;落っ
ことした
    if nol<=0:bmax++:gosub *fadeout:nol=10:goto *init    ;次の面へ
    await 15                    ;待つ（速さの調節はここだよ）

    goto *main
*explan                         ;簡易ヘルプ
    palcolor 202
    boxf 0,0,300,400
    palcolor 201
    font "MSゴシック",18
    pos 10,10
    mes {"
    --LINE  BALL--

    <説明>

    バーの移動　　マウス

    バーで打つごとに１点追加、
    １０点になると次の面に行けます。

    コツはバーの近くにマウスを持って
    いく事かな？




                    <<Mouse Left Button>>
    "}
*@
    stick keys
    if (keys&16)|(keys &32)|(keys &256):goto *ti
    wait 1
    goto *@b

*g                              ;画像描画
    pos 0,0
    gcopy 2,0,0,300,400         ;背景転送
```

★1月2月と続けてサターンBASICの文字。復活せよとのサインか？／自分で作れと言われそうだが，ジョイスティック→USB　買いて～。本命はサターン2USBだが。（福島市・**南斗オレガ26歳**）…【**影**：最後に告白！本命はつぐ美ちゃんです。**つぐ美**：ごめんなさい。】

```
	palcolor 255			;白
	pos 5,5:mes ""+bmax+"角形"
	pos 200,0:mes "Score:"+scr
	mes "Next :"+nol

	if re>0{			;残機表示
	repeat re
	pos cnt*20+20,380
	gcopy 2,103,400,17,3
	loop
	}

	pos ball.0.0,ball.0.1		;頂点を結ぶ
	repeat bmax
	line ball.cnt.0,ball.cnt.1
	loop
	repeat bmax			;頂点画像転送
	pos ball.cnt.0-1,ball.cnt.1-1
	gcopy 2,100,400,3,3
	loop

	line ball.0.0,ball.0.1		;一周させる

	pos mousex-(bl/2),300		;バーの位置
	gcopy 2,0,400,100,3		;バー画像転送

	return
*bmove					;頂点の動き
	out=0				;落ちた？
	repeat bmax
	ball.cnt.0+=ball.cnt.2
	if ball.cnt.0<0:ball.cnt.0=0:ball.cnt.2=ball.cnt.2*-1:snd 1
	if ball.cnt.0>300:ball.cnt.0=300:ball.cnt.2=ball.cnt.2*-1:snd 1
				;↑頂点x座標の動き
				;↓頂点y座標の動き
	ball.cnt.1+=ball.cnt.3
	if ball.cnt.1<0:ball.cnt.1=0:ball.cnt.3=ball.cnt.3*-1:snd 1
	if (ball.cnt.1>300)&(ball.cnt.1<305){

	bx=mousex-(bl/2)
	kk=bx+bl
		if (bx<ball.cnt.0)&(kk>ball.cnt.0){	;打ち返し
			snd 0
			ball.cnt.1=300:ball.cnt.3=ball.cnt.3*-1
			rnd ball.cnt.2,7	;ランダムに打ち返す
			ball.cnt.2-=3		;
			rnd ball.cnt.3,3
			ball.cnt.3=ball.cnt.3*-1
			if ball.cnt.2=0:ball.cnt.2=1
			if ball.cnt.3=0:ball.cnt.3=-1
			scr++:nol--		;打ったので点追加
		}

	}

	if ball.cnt.1>400:out=1		;下に落ちた
	loop


	return
*fadein					;徐々に暗く
	redraw 1
	repeat 128
	palfade cnt*-2
	redraw
	await 10
	loop

return
*fadeout				;徐々に明るく
	redraw 1
	repeat 128
	palfade cnt*2
	redraw
	await 10
	loop

return

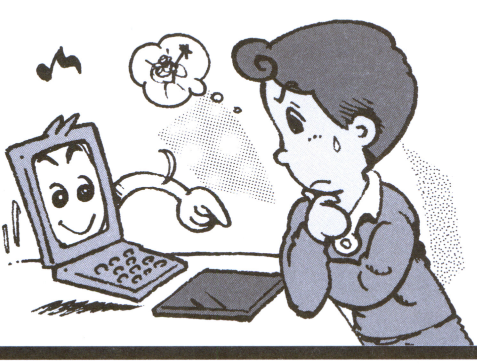
return

*over					;ゲームオーバー
	gosub *fadeout
	palfade 0
	redraw
	palcolor 255	;白
	font "MSゴシック",35
	pos 10,130
	mes "Your Score is "+scr


*@
	stick keys
	if keys&256:goto *st
	wait 10
	goto *@b
stop
```


★最近，押入れを整理してるとファミリーベーシックが…懐かしくて遊んでます。（北海道空知郡・WILDWOLF21歳）…**【Dr.D：ほう，無事動いたわけじゃな。編：ファミリーベーッシクを使った「ファミコン・プログラム大全集」なんての作ったっけなー。懐かしい！】**

HSP V2.55



かっこいい○×ゲーム
すりーでぃーまるばつ
3D○×

Wind Voice（28歳）

内容

みんな知っている○×ゲームです。コンピュータと対戦します。プレイヤーは先手/後手の選択（**写真1**）とコンピュータの強さ（3段階）を指定できます。コンピュータは○×ゲームとしては最強アルゴリズムなので，「最強モード」ではプレイヤーは引き分けることしかできなくなってしまいました。

遊び方

ルールはあえて説明するまでもないと思います。起動したあと，指示に従って先手か後手かを選び，コンピュータの思考ルーチンの強さを選ぶとゲームが始まります。プレイヤーは先手後手にかかわらず○で，コンピュータが×です。

入力はテンキーで行いますが，テンキーのない環境のためにj,k,l,u,i,o,7,8,9を使えるようにしてあります（**写真2**）。

リスト入力の注意

コンパイル後（デバッグモードではなく）実行するには，実行ファイルと同じフォルダにHSP2.55に付属している3Dグラフィック拡張モジュールhgimg.dllが必要です。

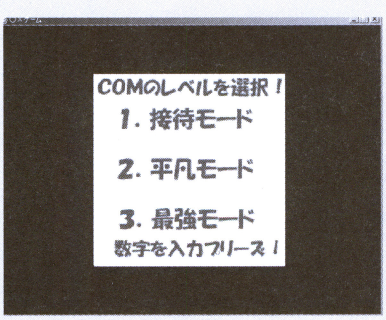
こだわり

今回は下記の点にこだわりました。
①あえて3D画像を使うこと（3Dゲームの習作として作成しました）
②思考ルーチンをしっかり作ること
③リストはコメントを含めて10000バイト以内

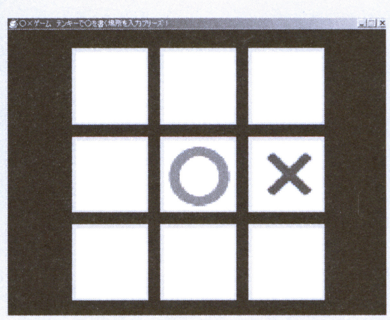
苦労した点

テクスチャ情報つきMXファイルの作成にかなりの試行錯誤が必要でした。思うに，この煩雑さのためにHSPの3Dゲームの投稿が少ない（ない？）のではないでしょうか。以前ベーマガでもHSPで3Dゲームを作る記事があったように思いますが，残念なことに保存していませんでした。テクスチャつきMXファイルはフリーでは作れないのでしょうか……

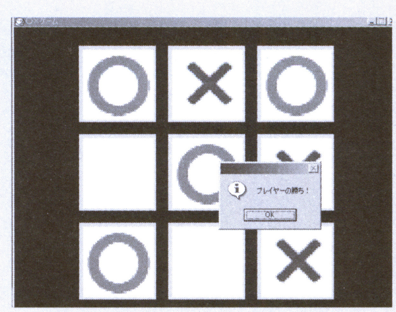
HSP作者のおにたまさんは，
①LightWaveでDirectXのオブジェクト



《写真1》はじめに先攻後攻と，モードを設定する



《写真2》ゲームは普通の○×ゲーム



《写真3》なんとかCPUに勝利

★4月から世間でいう社会人です。某ゲーム会社で働くことになっているのですが，どうなることやら…。（神奈川県横浜市・おぐら21歳）……【編：就職おめでとうございます。ところで，どこの会社？もしかすると，これから仕事で付き合うことになるかもね。】

であるXファイルを作成
②おにたまさん製ソフトXtoMXを使用してMXファイルを作成
という手順を紹介していますが，LightWaveはプロ向けであり，私を含め多くの「アマグラマー」には高価すぎると思います。

結局，さまざまなソフトを試した結果，DoGAが安価にテクスチャ付きXファイルを作成できますので，これとXtoMXを組み合わせて作成しました。しかし，こうしてやっとできたファイルもゲーム画面で動かしてみると（変換前は正常だったのに）テクスチャが裏返しに描画されてしまい文字が読めない，ということがおこりました。これは画像ファイルの一部を左右反転することでごまかしています。画像作成にはフリーソフトPixiaを使用しました。

MXファイルさえ，できてしまえば半分完成したようなものでした。しいて言えば思考ルーチンに頭を使いましたが，これは楽しい作業でした。自分で言うのもなんですが，非常にコンパクトにまとまっていい出来だと思います。
思考データなどに配列を多用していますが，コメントを見ればざっくりしたところはわかっていただけるのではないでしょうか。

改造

思考ルーチンはこれ以上強くならないはずです。人 vs 人，COM vs COM対戦を作るなどはどうでしょうか。あとは3D画像ならではのアクションを増やしてみるなどがあります。ゲーム中カメラが一切動いていませんので，このあたりも改造のしがいがあるかもしれません。

作者Webページ
風楼：http://windvoice.tripod.co.jp/

★使ったソフトなど

効果音：CK GAME FACTORY様
http://tokyo.cool.ne.jp/cgf/
CG作成：Pixia
http://homepage2.nifty.com/~maru_tacmi/
オブジェクト作成：DoGA
http://www.doga.co.jp/ptdoga/

CHECKER FLAG

影：単純なゲームとはいえ、演出がかっこいいゾ！
Dr.D：それと細かいことじゃが，ゲーム終了後のメッセージボックスはやめた方がよいな。タイトル画面と同じ，画像ファイルを用意すべきじゃ！
編：リストは長くないのですが，データが多いために超大作扱いとさせていただきました。
つぐ美：あと，住所は間違えないでくださいね。

Pick up LIST

《リスト》コンピュータ思考ルーチン

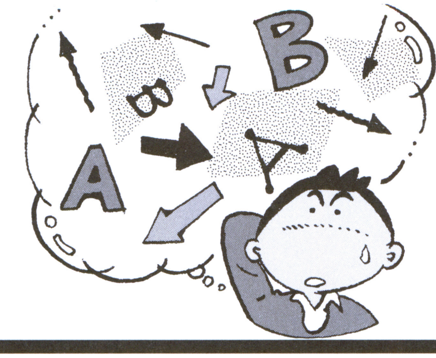
```
;--------------------------------------------
	; コンピュータの番
	;--------------------------------------------
*comturn
	;思考ルーチンはこのrepeat～loopです。
	repeat 29
		if lv>com_lv.cnt {
			x1=com_x1.cnt : x2=com_x2.cnt : x3=com_x3.cnt
			y1=com_y1.cnt : y2=com_y2.cnt : y3=com_y3.cnt
			if map.x1.y1+map.x2.y2+map.x3.y3=com_sum.cnt {
				if map.x1.y1=0 : X=x1 : Y=y1 : goto *cput
				if map.x2.y2=0 : X=x2 : Y=y2 : goto *cput
				if map.x3.y3=0 : X=x3 : Y=y3 : goto *cput
			}
		}
	loop

*@
	;決まらなければランダムに取る
	rnd X,3 : rnd Y,3
	if map.X.Y!0 : goto *@back
```

★あくせすもOB，ナイスSHORT！川相オチ？／春休み，金欠発病，金祭。寝る子は育つ！／250号おめでとうございます。それでは次回までサヨナラ。（愛知県名古屋市・**最馬場可？歳**）……【**つぐ美：**相変わらずはじけていますね。**影：**私もついていけないざます。】

ベストプログラム

PIECE用



素早くボタンを押せ
ぷれす60 Version1.12

ていこく（23歳）

PIECE

ゲームの内容

上下左右，それにAボタン，Bボタンの6種類のボタンを使って，画面上にある60個のマークをすべて消し，それに要した時間を競うゲームです。また，早く消せば消すほどコンボとなり，高得点が得られます。

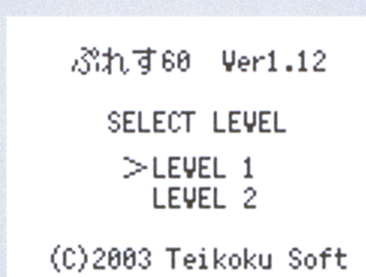
入力方法

(1)リスト（prog.c）を打ち込んで保存します
(2)MS-DOSプロンプトを起動し，保存したファイルがあるディレクトリに移動します
(3)"pcc33 prog.c"と入力し，リターンキーを押します
(4)"run"と入力し，リターンキーを押せば実行となります

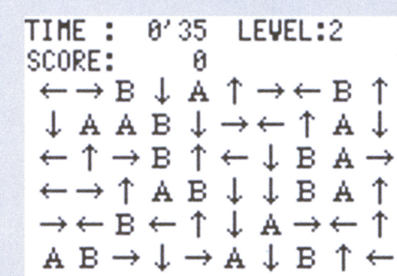
実行するとタイトル画面が表示されます（**写真1**）。上下でレベルを選択し，STARTでゲームが始まります。

遊び方

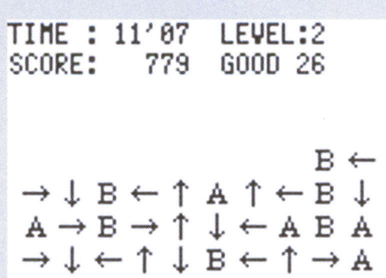
レベルは，ゲーム時に登場するマークの種類の数に関係してきます。ゲームを始めると，カウントダウン表示の後，合計60個のマークが画面に現れますので，左上から右に向かってマークを消していってください（**写真2**）。マークを消すには，マークと同じボタン（↑↓←→AB）を押してください。マークを消したとき，素早く消していくと，画面右上に



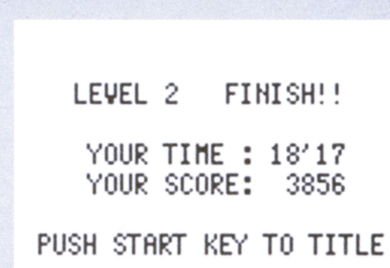
《写真1》タイトル画面で，レベルが選択できる



《写真2》ゲーム開始直後



《写真3》素早く，でも正確に



《写真4》ゲーム終了。2000点を目指そう！

「GREAT」または「GOOD」と表示されます（**写真3**）。連続で素早く消していくとコンボとなり，通常より多く点数が加算されていきます。右端まで行くと，下の段の左端に移動します。

こうして，右下端のマークまで消すとゲームが終了し，スコアとタイムが表示されます（**写真4**）。ここでSTARTボタンを押すとタイトル画面に戻ります。ゲームを終了させるには，STARTとSELECTを同時に長く押してください。レベル1ではマークは↑↓←→の4種類，レベル2では↑↓←→ABの6種類となります。

作成後記

ベーマガ2002年5月号のOFで「PIECEプログラムができたら送ります」と言っておきながら，今まで投稿しませんでした。申し訳ありませんでした。

今回の「ぷれす60」は，シンプルですが，それゆえ最速＆最高得点を狙うと，はまると思います。とりあえず，タイムで18秒台，得点で2000点を目指してください。**リスト**は300行ちょっとでそれほど長くないと思います。ひととおりコメントを付けてあるので，参考にしてみてください。それから，ゲームスタート時のカウントダウン部分の追加を提案して下さったヅラChuさん，どうもありがとうございました。今度はSLGを作ってみたいですね。もちろん，大作は気力が続かないので小規模のものになりますが。

私のメールアドレスとHPアドレスは
ymd3106@jomon.ne.jp
http://www.jomon.ne.jp/~ymd3106/
です。また，トップページから自作のPIECEのソフトを展示している「52PIECE!」に移動できますので，PIECE作品もどうぞご覧下さい。感想お待ちしています。HPにも遊びに来てください。

CHECKER FLAG

影：よし，ひたすら画面見ないで連打してやるぜ！
編：そうじゃなくて，ちゃんと見ながら正確に押した方が点が高いみたいですよ。ほら。
Dr.D：ただ連打するのではなくて，プレイヤーの判断力を要求するゲームのようじゃな。プログラムの方も，きれいにまとめてあっていいゾ！

《第1表》変数表

setkey[60]	マークのデータ（↑,↓,←,→,A,Bがそれぞれ0,1,2,3,4,5）
score	スコア
combo	コンボ数
level	レベル
ptime	前回キーを押したときの時間
cnt	カウンタ(タイトル画面や結果画面で使用)
lkey	キーパッドの情報用
setpos	現在位置(0-59)
lastkey	直前に押されたボタン（押しっぱなし防止用）
gamemode	ゲームの状態
t0,t1,t2	時間用
i,j	ループ用
a,b,t	乱数用
その他	汎用

《リスト》ぷれす60のプログラム・リスト

```
//////////////////////////////////////
//      ぷれす60 Version 1.12       //
//                                  //
//      Presented By ていこく        //
//        完成日：2003/01/12         //
//////////////////////////////////////

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#include <piece.h>

// ゲーム進行用の定数を設定
#define GAMEINIT  0
#define GAMEMAIN  1
#define GAMEEND          2
#define GAMETITLE 3
#define GAMESEL          4
#define GAMECOUNT 5

unsigned char vbuff[128*88];
static unsigned char draw;

// 変数設定
char keyname[6][3] = { "↑","↓","←","→","A","B" };
int setkey[60],score,combo,level,ptime;
int cnt,lkey,setpos,lastkey,gamemode,t0;

// ゲーム開始前のカウントダウン
void myGameCountDown ( void ) {
int t1;
char c[4];
        t1=pceTimerGetCount()-t0;
        if (t1<3300) { sprintf( c, "Go!¥0"); }
        if (t1<3000) { sprintf( c, " 1 ¥0"); }
        if (t1<2000) { sprintf( c, " 2 ¥0"); }
        if (t1<1000) { sprintf( c, " 3 ¥0"); }
        pceLCDPaint( 0,0,0,128,88 );
        pceFontSetType(1);
        pceFontSetPos(52,36);
        pceFontPrintf("%s",c);
        if (t1>3300) { gamemode=GAMEINIT; }
```

★私のペンネームは「クレナイ　ロクチュウシン」と読みます。「紅六さん」はやめてください。（京都府長岡京市・**紅六忠臣25歳**）……【**影**：これは失礼した。ゆるせ，クレナイ殿。ところで，ワシのことを「エイさん」と呼ぶ輩がいるようじゃが，「カゲ」と呼べい。】

```
}

// ゲーム開始前の準備
void myGameInit( void ) {
int i,j,a,b,t,setpat[6];
  // マークを60個並べる
          for (j=0;j<( 60/(level*2+4) );j++) {
            for (i=0;i<=5;i++) { setpat[i]=i; }
                    for (i=0;i<20;i++) {
                              a=i % (level*2+4); b=rand() % (level*2+4);
                              t=setpat[a]; setpat[a]=setpat[b]; setpat[b]=t;
                    }
                    for (i=0;i<(level*2+4);i++) {
                              setkey[j*(level*2+4)+i]=setpat[i];
                    }
          }

  // マークを画面上に表示する
          pceLCDPaint( 0,0,0,128,88 );
          pceFontSetType(0);
          pceFontSetTxColor(3);
          for (i=0;i<=59;i++) {
                    pceFontSetPos((i % 10)*12+4, (i / 10)*11+20 );
                    pceFontPrintf("%s",keyname[setkey[i]]);
          }

  // レベル表示
          pceFontSetPos(70,0);
          pceFontPrintf("LEVEL:%1d",(level+1));

  // 変数の初期化
          setpos=0; lastkey=-1;
  t0=pceTimerGetCount(); ptime=t0;
          score=0; combo=0;
}

// ゲームのメインルーチン
void myGameMain( void ) {
int cmd,t1,t2;

  // どのボタンが押されているか
  lkey = pcePadGet();

  // 押されているボタンによってcmdの内容を変える（押されてないと-1）
  cmd=-1;
```

```
        if (lkey & PAD_UP) { cmd=0; }
        if (lkey & PAD_DN) { cmd=1; }
        if (lkey & PAD_LF) { cmd=2; }
        if (lkey & PAD_RI) { cmd=3; }
        if (lkey & PAD_A) { cmd=4; }
        if (lkey & PAD_B) { cmd=5; }

  // 押しっぱなしは押していないのと同じ
  if (cmd != lastkey) { lastkey=cmd; } else { cmd=-1; }

  // ボタンが表示されているマークと同じか
  if (setkey[setpos] == cmd) {

    // マークを消す
          pceFontSetPos( (setpos % 10)*12+4, (setpos / 10)*11+20);
                    pceFontPrintf("  ");

    // 前回マークを消した時間からどれくらい経過してるか
                    t1=pceTimerGetCount()-ptime;
                    ptime=pceTimerGetCount();

    // 0.3秒以内なら"GREAT"表示でスコア、コンボ増加
                    if (t1<300) {
                              combo++;
                              pceFontSetPos(70,10);
                              pceFontPrintf("GREAT%2d",combo);
                              score=score+(combo*3);
                    } else {

      // 0.6秒以内なら"GOOD"表示でスコア、コンボ増加
                              if (t1<600) {
                                        combo++;
                                        pceFontSetPos(70,10);
                                        pceFontPrintf("GOOD %2d",combo);
                                        score=score+combo;
                              } else {

        // 0.6秒以上だったら表示なし、コンボリセット
                                        pceFontSetPos(70,10);
                                        pceFontPrintf("       "); // 半角スペースが7つ
                                        combo=0;
                                        score=score+1;
                              }
                    }
```


★こんにちは！ベーマガ買い始めてちょうど1年！でも初投稿です！（神奈川県横浜市・masanaka7 13歳）……**【影：初投稿が最終号掲載とは，記念になったんじゃないですか？復刻の可能性も残しつつ，またどこかでお会いしましょう。編：みんな，がんばってね。】**

```
    // 判定するマークの位置を移動する
    setpos++;

    // 右下端のマークが終わったら結果画面へ
            if (setpos>=60) { gamemode=GAMEEND; cnt=0; }
  }

  // スタートからの経過時間を算出
        t1=pceTimerGetCount()-t0;
  t2=(t1 % 1000)/ 10; t1=(t1 / 1000);

  // スタートからの経過時間を表示
        pceFontSetPos( 0, 0);
        pceFontPrintf("TIME :%3d'%02d",t1,t2);

  // スコアの表示(コンボが続くほど点数が高くなる)
        pceFontSetPos( 0, 10);
        pceFontPrintf("SCORE:%6d",score);
  draw=1;
}

// ゲーム終了時の結果画面
void myGameEnd( void ) {
int t1,t2;
        switch (cnt) {

  // 最初は結果画面を表示する
        case 0:
        pceLCDPaint( 0,0,0,128,88 );
                t1=pceTimerGetCount()-t0;
        t2=(t1 % 1000)/ 10; t1=(t1 / 1000);
                pceFontSetType(0);
                pceFontSetTxColor(3);
                pceFontSetPos(19,20);
                pceFontPrintf("LEVEL %d   FINISH!!",(level+1));
                pceFontSetPos(24,40);
                pceFontPrintf("YOUR TIME :%3d'%02d",t1,t2);
                pceFontSetPos(24,50);
                pceFontPrintf("YOUR SCORE:%6d",score);
                pceFontSetPos(7,70);
                pceFontPrintf("PUSH START KEY TO TITLE");
                cnt=1;
                break;

  // 結果画面の直前からSTARTボタンが押されていたら先に進まない
```

```
  // (これを許すと結果画面が見えずにタイトル画面に進むため)
        case 1:
                lkey = pcePadGet();
                if (lkey == 0) { cnt=2; }
                break;

  // STARTボタンが押されていたらタイトル画面へ
        case 2:
          lkey = pcePadGet();
                if (lkey & TRG_C) { gamemode=GAMETITLE; cnt=0; }
        }
}

// ゲームのタイトル画面
void myGameTitle( void ) {
int i;
        switch (cnt) {

  // 最初はタイトル画面を表示する
        case 0:
        pceLCDPaint( 0,0,0,128,88 );
                pceFontSetType(0);
                pceFontSetTxColor(3);
                pceFontSetPos(22,10);
                pceFontPrintf("ぷれす60  Ver1.12");
                pceFontSetPos(34,30);
                pceFontPrintf("SELECT LEVEL");
                pceFontSetPos(49,45);
                pceFontPrintf("LEVEL 1");
                pceFontSetPos(49,55);
                pceFontPrintf("LEVEL 2");
                pceFontSetPos(14,75);
                pceFontPrintf("(C)2003 Teikoku Soft");
                level=0; cnt=1;
                break;

  // タイトル画面の直前からSTARTボタンが押されていたら先に進まない
  // (これを許すとタイトル画面が見えずにゲーム画面に進むため)
        case 1:
                lkey = pcePadGet();
                if (lkey == 0) { cnt=2; }
                break;

  // 上下ボタンでレベルを選択、STARTボタンでゲーム画面へ移動
        case 2:
```


★こっ，4月号には読者プレゼントがないっ！　何かの間違いではないかと思った。けど，やっぱりなかった(°o°)。(広島県廿日市市・NSR250R＋横浜ナンバー39歳）……【編：ごめんなさい，今月号もプレゼントはありません。影：代わりに私の笑顔を差し上げます。】

```
			lkey = pcePadGet();
			if (lkey & TRG_UP) { level=0; }
			if (lkey & TRG_DN) { level=1; }
			if (lkey & TRG_C) { gamemode=GAMECOUNT; t0=pceTimerGetCount(); }

	// レベル選択のカーソルを表示
			pceFontSetPos(39,(1-level)*10+45);
			pceFontPrintf("  ");
			pceFontSetPos(39,level*10+45);
			pceFontPrintf(">");
		}
}

/////////////////////////////////////////////
//        ここからPIECE本体の処理          //
/////////////////////////////////////////////

// 実行時最初に来るのはここ
void pceAppInit( void )
{
		PCETIME strTime;

		pceTimeGet( &strTime );
		srand( strTime.s100 + strTime.ss * 100 + strTime.mi * 6000 );

		pceLCDDispStop();
		pceLCDSetBuffer( vbuff );

  // インターバルは0.01秒
  pceAppSetProcPeriod( 10 );
		memset( vbuff, 0, 128*88 );

  draw = 1;

  // 最初はタイトル画面に移動するようにする
  gamemode=GAMETITLE; cnt=0;

		pceLCDDispStart();
}

// PIECE本体のメインルーチン(0.01秒ごとに呼び出される)
void pceAppProc( int cnt )
{
  // gamemodeの内容によって、飛び先を指定する
		switch (gamemode) {

  // gamemode = GAMETITLE ならタイトル画面へ
		case GAMETITLE:
				myGameTitle();
				draw=1;
				break;

  // gamemode = GAMEINIT ならゲーム開始の準備画面へ
  case GAMEINIT:
				myGameInit();
				draw=1;
				gamemode=GAMEMAIN;
				break;

  // gamemode = GAMEMAIN ならゲームのメインへ
		case GAMEMAIN:
		  myGameMain();
				draw=1;
				break;

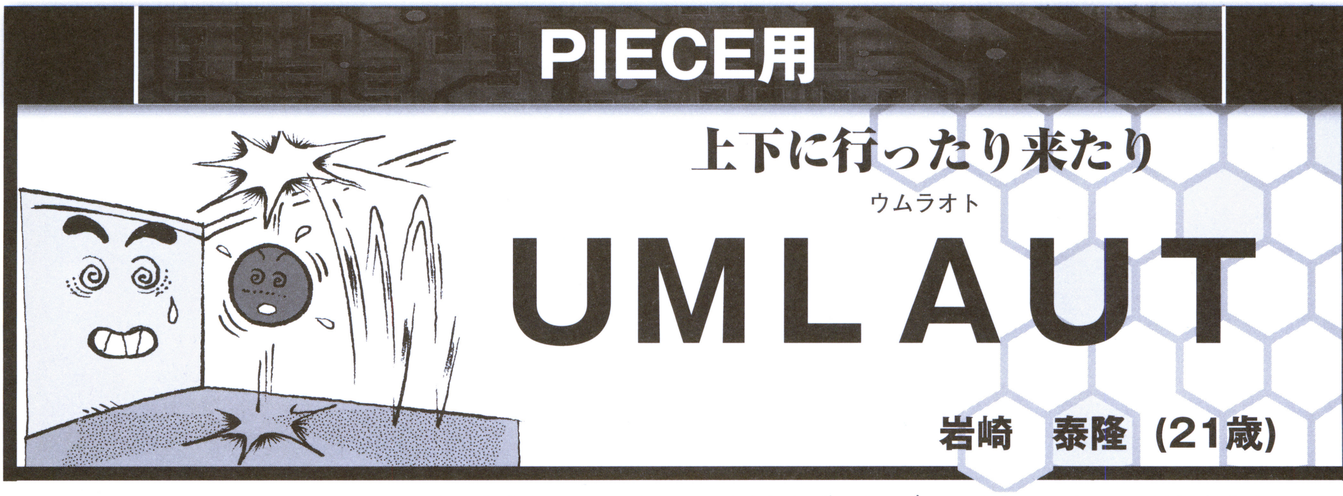
  // gamemode = GAMEEND なら結果画面へ
		case GAMEEND:
				myGameEnd();
				draw=1;
				break;

		// gamemode = GAMECOUNT ならゲーム前のカウントダウン画面へ
		case GAMECOUNT:
				myGameCountDown();
				draw=1;
				break;
		}

  // 画面に表示
		if ( draw ) {
				pceLCDTrans();
				draw = 0;
		}
}

// START+SELECT同時押しでここに来る
// （何も記述しないので、そのままプログラムを終了させる）
void pceAppExit( void )
{
}
```


★僕のゴロパソ日記：今日はVBでプログラミング。でも，プログラムを作るより，バグ取りに時間がかかった(泣)。(神奈川県小田原市・**漆畑龍典14歳**)……**【Dr.D：**プログラミングで一番大変で，大事なのが，動作チェック！バグ取りがまた上達につながるぞ。**】**



上下に行ったり来たり UMLAUT（ウムラオト）

岩崎　泰隆（21歳）

はじめに

この作品は，マイコンBASICマガジン1997年９月号掲載，ゲル・ピン氏によるPC-8801mkIISR用ゲーム「UMLAUT」の移植作品となっております。末期の88のゲームということで，ご存じない方も多いかと思われますが，はっきりいってこのゲームなかなかおもしろいので，あの興奮をもう一度と移植してみました。ベーマガの旧きよき投稿プログラムの感じを，ぜひ堪能してみてくださいませ～。

入力方法

まず，適当なエディタなどで**リスト１**を入力しmain.cという名で保存。そのまま続けて，**リスト２**を入力しchara.cという名で保存。そして，sysdev¥music¥instdef.cというファイルをコピーして同じフォルダに入れて，コマンドラインからpcc33 main.c chara.c instdef.cと入力してコンパイルしてくれれば，あとはできあがったsrfファイルを実行するだけです。

遊びかた

タイトル画面（**写真１**）でSTARTを押せば始まります（**写真２**）。

プレイヤーは，画面で上下運動を行っているボールで，これはパッドの左右で動かせます（**写真３**）。そしてこのボールを画面上部および，下部に敷き詰められている同色のボールに当てることにより，得点と残りタイムを得ることができます。同色のボールに何回か当てるとレベルが上がり，レベルが上がることによって，ボールの移動速度が上がったりブロックの数が増えたりします。

もし，違う色のボールに当ててしまった場合は，その色がプレイヤーの色となりますが，誤ってブロックに当ててしまうとゲームオーバーとなってしまうので注意してください（**写真４**）。

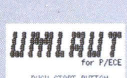
なお，ハイスコアは保存するので，このプログラムとは別にもう１セクタだけフラッシュメモリを空けておいてください。空きがない場合，ハイスコアは保存されません。

プログラム

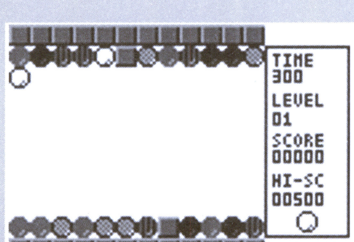
リスト掲載を目指して，かなり詰め込んでいる箇所や，ごちゃごちゃしている箇所が多数見受けられますが，頑張って解析してみてください。その代わりといってはナンですが，コメント類はできるだけたくさん入れておいたので…。

音楽や効果音は，原作をそのまま流用しています。ただ，原作のサウンドを録音して，それを再生となると容量的に厳しいので，すべてMMLで書き直した物をmuccで変換して，これをリストに組み込んで音楽として再生しています。このことより，原作とは音色が異なります。

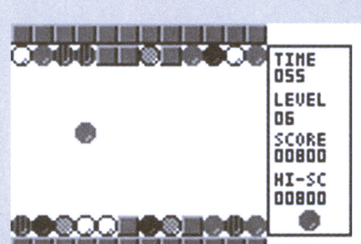
しかしながらPIECEの仕様として，PlayMusic()を頻繁に呼び出すと暴走す



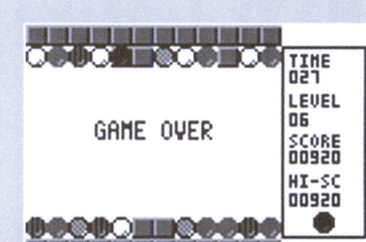
《写真１》テキストを重ね合わせたタイトル画面



《写真２》ゲームスタート



《写真３》同色のボールに当てよう



《写真４》ブロックに当たるとゲームオーバー

★のび太「ドラえもーん，いろいろ困ったよー」，ドラえもん「お金印刷機―！」（福岡市・前田宣弘29歳）……【つぐ美：私も，ドラえもんとお友達になりたーい。影：呼ばれて飛び出てジャジャジャジャーン！つぐ美：それって違うと思うんですけど。影：…。】

ることがあるらしいので，これを回避するためにレベルアップ直後に効果音が入らないことがあります。バグではないので報告されても直せません。どーしても報告したいのならば，アクアプラスの方へ。

改造方法

PIECEユーザー必須アイテムその２に登録されている（のか！？），『P/ECEハンドブックvol2』の95ページに書かれている，必要な音色だけリンクする方法を適用することができます。今回，利用している音色は@1(&i_saw0)のみなので，instdef.cを書き換えることによりファイルサイズを大幅に削減できます。詳しくは同書を参考にしていただければ。

あとは，レベルアップのタイミングやスコアの入り方など，自分好みに改造してみてください。わりと適当に作ったやつをそのまま採用してしまった感じなので、人によっては不満がでてくるかもしれないので。

最後に

原作をそのまま移植すると，ボールの動きが速くなったときに残像が残りまくりでゲームにならなかったので，この辺の調整にかなり苦労した感じです。いくらかルールを変更して，やっと遊べる作品になりました。うーん，88でもアナログディスプレイだったんで，残像云々ということを考える必要がなかったのはよかったなぁ。PIECEの性能悪すぎる液晶は，この手のアクションゲームには向いていないので，これはなんとかしてほしいかも（今更か）。

それとPIECEって（これに限らず，今のプログラム環境全般にいえることだけど），もう少し効果音を簡単に鳴らせるといいと思うのは自分だけではないはず。それこそ，昔のBASICってのは，PLAY"t177@23o5c8"とかで非常に手軽に音が出せたのに，なんで今のプログラム言語は，こう簡単にできないのかなー。

あと，要望感想質問随時承り中です。http://www002.upp.so-net.ne.jp/ic_yas/でお気軽にどうぞ。ではまた。

■参考にしました

・マイコンBASICマガジン1997年９月号　UMLAUT（ゲル・ピン氏）

・P/ECE HAND BOOK vol.2　著者：まかべひろし（発行：JC2　発売：電波新聞社）

CHECKER FLAG

つぐ美：PC88という旧機種の投稿作品をうまくPIECEへ移植してあります。PIECEの発色数が限られた液晶でありながら，ボールの色わけを工夫して，より多くのボールを表示しているわ。

編：誌面では伝わらないのですが，BEEP音みたいな効果音が面白いです。やろうと思えばかなりキレイな効果音が出せるPIECEで，あえてこういった音を出すのも，味がありますね。

Dr.D：リストで面白い点と言えば，タイトル画面のテキスト表示じゃな。思えば昔の投稿作品は，こういったテキスト表示を限界まで活用していたものじゃ。

《リスト１》main.cのプログラム・リスト

```
//----------------------------------------------------------------
//
//          UMLAUT for P/ECE
//
//  Copyright(C) 1997 T.I / 2003 NISHISAITAMA PROJECT all rights reserved.
//----------------------------------------------------------------
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <piece.h>

//    定数定義（リスト軽量化のために仕方なくやっています...）
#define P08 Player.x/8
#define P72 Player.y>=72

//    関数プロトタイプ宣言
void GameInit();
void GameMain();
void GameOver();
void GameTitle();
void MainLoop();
void BltGraphics(int x,int y,PIECE_BMP *src,int sx,int sy,int w,int h);
void ScreenDraw();
void Swap(int *x,int *y){
    int tmp=*x;*x=*y;*y=tmp;     //    ＢＡＳＩＣのＳＷＡＰ文と同じ
}

//    グローバル変数定義
unsigned char vbuff[128*88];     //    この辺はチュートリアル参照
extern unsigned char CHARA[];

DRAW_OBJECT    obj;
PIECE_BMP      pBmpChara;

int     gameState;  //    ゲームの状態
BOOL saveCheck;     //    ハイスコアを更新するかどうか
int blockX[2][5];   //    ブロックのX座標
int blockMax;       //    ブロックの数
int blockCount;     //    ブロックが増えるまでのカウント

//    構造体宣言
struct TPLAYER{
```

★春は影さんの季節ですね？（三重県四日市市・**ぶんちょ28歳**）……【**影：**どの季節だって，私の季節だ！ぐはははは。ま，春が一番似合ってるかな，この私には……ぐー。**つぐ美：**はじまった，すぐ寝るんだから，もう。**編：**寝てるか，邪魔してるかの影さんです。】

```
    int x,y;        //    座標
    int vx;         //    左右移動の微調整
    int d;          //    進む向き
    int c;          //    プレイヤーの色
    int time;       //    残り寿命
    int score;      //    スコア
    int highScore;  //    ハイスコア
    int speed;      //    上下運動の速度
    int next;       //    レベルアップまでの数
}Player;

struct TBALL{
    int c;    //    ボールの種類
}Ball[2][16];

//    ＢＧＭ関係
unsigned char startBgm[]={
    0x00,0x06,0x25,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xe7,0xb1,0xe8,0x01,0x0c,0xb0,0xb4,
    0xb7,0xb4,0xb7,0xbb,0xb7,0xbb,0x60,0xbc,0x00,0xe3,0x15,0x00,0x01,0x00,
};
unsigned char levelUpBgm[]={
    0x00,0x06,0x22,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xe7,0xb1,0xe8,0x01,0x06,0xb7,0xb9,
    0xbb,0xb9,0xbb,0x18,0xbc,0x00,0xe3,0x15,0x00,0x01,0x00,
};
unsigned char overBgm[]={
    0x00,0x06,0x2f,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xe7,0xb1,0xe8,0x01,0x0c,0xbc,0xe0,
    0xbc,0xbc,0xbe,0xc0,0xc1,0xe0,0xc1,0xc1,0xc0,0xbe,0xbc,0xe0,0xbc,0xbc,
    0xbc,0xbc,0x30,0xbc,0x00,0xe3,0x15,0x00,0x01,0x00,
};
unsigned char hitBgm[]={
    0x00,0x06,0x1e,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xe7,0xb1,0xe8,0x01,0x06,0xb7,0x0c,
    0xb2,0x00,0xe3,0x15,0x00,0x01,0x00,
};
unsigned char missBgm[]={
    0x00,0x06,0x1c,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xe7,0xb1,0xe8,0x01,0x0c,0x9f,0x00,
    0xe3,0x15,0x00,0x01,0x00,
};
//------------------------------------------------------------------------
```

```
//    ゲームの初期設定
//------------------------------------------------------------------------
void GameInit()
{
    int i,j;

    //    変数初期化
    Player.x=0,Player.y=16,Player.d=1,Player.c=rnd(5)+1;
    Player.time=300,Player.score=0;
    Player.speed=3,Player.next=6;

    blockMax=1;
    blockCount=12;

    for (j=0;j<2;j++){
        for (i=0;i<5;i++) blockX[j][i]=13;    //    よけいなブロックは全部画面外に
        for (i=0;i<12;i++) Ball[j][i].c=rnd(5)+1;

        blockX[j][0]=rnd(12);
        Ball[j][blockX[j][0]].c=0;    //    一つだけブロック作成
    }

    //    画面右側作成
    pceLCDPaint(3,96,8,32,72);pceLCDPaint(0,97,9,30,70);
    pceFontSetType(2);
    pceFontSetPos(99,11);pceFontPrintf("TIME");
    pceFontSetPos(99,26);pceFontPrintf("LEVEL");
    pceFontSetPos(99,41);pceFontPrintf("SCORE");
    pceFontSetPos(99,56);pceFontPrintf("HI-SC");

    ScreenDraw();
    BltGraphics(Player.x,Player.y,&pBmpChara,Player.c*8,0,8,8);

    PlayMusic(startBgm);    //    はじめの音楽再生
    pceLCDTrans();
    while (MusicCheck()==0);    //    曲再生中は停止

    gameState=1;    //    ゲームスタート
}
//------------------------------------------------------------------------
//    ゲームメイン
//------------------------------------------------------------------------
void GameMain()
```

★あと6号，10月号で100号か…。もちろん，ベーマガ読者なら，わかりますよね！？（青森県南津軽郡・ていこく23歳）……【編：256号では何かやるんですか？なんていうお便りもありましたが，256号もちませんでした。ごめんなさい。影：251号で打ち止めです。】

```
{
    int a,i;
    static int playAllowSound=0;

    //    残りタイムが減り、0になったら終わる
    Player.time--;
    if (Player.time==0) gameState=2;

    //    立て続けに音楽を再生できないようにする
    if (playAllowSound>0) playAllowSound--;

    //    キー入力と移動
    Player.vx+=(pcePadGetDirect()&PAD_RI)-(pcePadGetDirect()&PAD_LF);
    if (Player.vx<-2) Player.x-=8,Player.vx=0;
    if (Player.vx> 1) Player.x+=8,Player.vx=0;

    Player.y+=Player.d*Player.speed;

    if (Player.x<0  ) Player.x=88;
    if (Player.x>88) Player.x=0;

    //    画面上部か下部にボールが到着
    if (Player.y<=8 || Player.y>=72){
        if (Ball[P72][P08].c==Player.c){
            a=rnd(blockMax);
            Ball[P72][blockX[P72][a]].c=rnd(5)+1;
            blockX[P72][a]=P08;
            Ball[P72][P08].c=0;    //    ここがブロックになる

            Player.time+=20;    //    ある程度タイムが復活

            Player.next--;blockCount--;
            Player.score+=(Player.speed+blockMax-3)*10;

            if (Player.score>Player.highScore)    //    ハイスコアも更新？
                saveCheck=TRUE,Player.highScore=Player.score;

            //    レベルアップ関係はここ
            if (Player.next==0 && Player.speed<7){
                Player.speed++,Player.next=10;
                playAllowSound=14;
                PlayMusic(levelUpBgm);
            }
            if (blockCount==0 && blockMax<5){
                blockMax++,blockCount=10;
                playAllowSound=14;
                PlayMusic(levelUpBgm);
            }

            //    上の音が再生直後の時は、ぶつかったときの音を念のため再生しない
            if (playAllowSound==0) PlayMusic(hitBgm);
        }else if (Ball[P72][P08].c==0){
            gameState=2;    //    ここはゲーム終了
        }else if (Ball[P72][P08].c!=Player.c){
            Swap(&Ball[P72][P08].c,&Player.c);    //    他の色の場合は入れ替えるだけ
            if (playAllowSound==0) PlayMusic(missBgm);
        }
        Player.d=-Player.d;    //    上下移動反転
    }

    if (Player.y<8) Player.y=8;if (Player.y>72) Player.y=72;    //    はみ出し判定

    //    画面描画
    pceLCDPaint(0,0,8,96,72);
    ScreenDraw();

    BltGraphics(Player.x,Player.y,&pBmpChara,Player.c*8,0,8,8);    //    自キャラ
}
//---------------------------------------------------------------------------
//    ゲームオーバー
//---------------------------------------------------------------------------
void GameOver()
{
    static BOOL init=TRUE;
    FILEACC fa;

    //    ゲームオーバー初期化
    if (init==TRUE){
        pceFontSetType(0);pceFontSetPos(25,35);pceFontPrintf("GAME OVER");
        init=FALSE;

        PlayMusic(overBgm);    //    ゲームオーバーの音楽再生
        pceLCDTrans();
        while (MusicCheck()==0);    //    曲再生中は停止

        if (saveCheck==TRUE){
```


★いよいよ新学期！今年から受験生です。プログラムする暇がない，いや，むしろ勉強する暇か？／フリーツール特集とかどうです？Blenderとか，Swaveとか，おもしろいです！ドット絵の講座とかも！（兵庫県朝来郡・INA17歳）……【つぐ美：がんばってね！】

```
        //    ハイスコアファイル更新
        pceFileCreate("umlaut.dat",sizeof(int));
        pceFileOpen(&fa,"umlaut.dat",FOMD_WR);
        pceFileWriteSct(&fa,&Player.highScore,0,sizeof(int));
        pceFileClose(&fa);

        saveCheck=FALSE;
      }
    }

    if (pcePadGet()&PAD_C) gameState=5,init=TRUE;
}
//----------------------------------------------------------------------
//    タイトル
//----------------------------------------------------------------------
void GameTitle()
{
    static BOOL ky=TRUE;

    //    STARTを押していたらはなすまで無効
    if (ky==TRUE && !(pcePadGet()&PAD_C)) ky=FALSE;

    memset(vbuff,0,128*88);

    //    最近ではあまり見られなくなったテキスト合成技
    pceFontSetType(0);pceFontSetBkColor(-1);
    pceFontSetPos(3,10);pceFontPrintf("# # ## ## #  ### # # ###");
    pceFontSetPos(2,17);pceFontPrintf("# # ##### #  # # # #  # ");
    pceFontSetPos(1,24);pceFontPrintf("# # # # # #  ### # #  # ");
    pceFontSetPos(0,31);pceFontPrintf("### # # # ## # # ###  # ");
    pceFontSetPos(3,10);pceFontPrintf("ﾛ ﾛ ﾛﾛ ﾛﾛ ﾛ  ﾛﾛﾛ ﾛ ﾛ ﾛﾛﾛ");//半角ｶﾅ
    pceFontSetPos(2,17);pceFontPrintf("ﾛ ﾛ ﾛﾛﾛﾛﾛ ﾛ  ﾛ ﾛ ﾛ ﾛ  ﾛ ");//のﾛです
    pceFontSetPos(1,24);pceFontPrintf("ﾛ ﾛ ﾛ ﾛ ﾛ ﾛ  ﾛﾛﾛ ﾛ ﾛ  ﾛ ");
    pceFontSetPos(0,31);pceFontPrintf("ﾛﾛﾛ ﾛ ﾛ ﾛ ﾛﾛ ﾛ ﾛ ﾛﾛﾛ  ﾛ ");

    pceFontSetPos(78,40);pceFontPrintf("for P/ECE");pceFontSetTxColor(rnd(4));
    pceFontSetPos(23,58);pceFontPrintf("PUSH START BUTTON");
    pceFontSetTxColor(3);

    if (ky==FALSE && pcePadGet()&PAD_C){
        ky=TRUE,gameState=0;
        memset(vbuff,0,128*88);
    }
    if (pcePadGet()&PAD_D) pceAppReqExit(0);    //    SELECTでゲーム終了
}
//----------------------------------------------------------------------
//    キャラクタ表示
//----------------------------------------------------------------------
void BltGraphics(int x,int y,PIECE_BMP *src,int sx,int sy,int w,int h)
{
    pceLCDSetObject(&obj,src,x,y,sx,sy,w,h,DRW_NOMAL);
    pceLCDDrawObject(obj);
}
//----------------------------------------------------------------------
//    画面作成
//----------------------------------------------------------------------
void ScreenDraw()
{
    int i,j;
    char szBuffer[50];

    //    画面上部および下部作成
    for (j=0;j<2;j++){
        for (i=0;i<12;i++){
            BltGraphics(i*8,j*64+8,&pBmpChara,Ball[j][i].c*8,0,8,8);
            BltGraphics(i*8,j*80,&pBmpChara,0,0,8,8);
        }
    }

    //    画面右側作成
    pceFontSetType(2);pceFontSetBkColor(0);
    pceFontSetPos(99,17);sprintf(szBuffer,"%03d",Player.time);pceFontPrintf(szBuffer);
    pceFontSetPos(99,33);sprintf(szBuffer,"%02d",Player.speed-
3+blockMax);pceFontPrintf(szBuffer);

pceFontSetPos(99,47);sprintf(szBuffer,"%05d",Player.score);pceFontPrintf(szBuffer);

pceFontSetPos(99,63);sprintf(szBuffer,"%05d",Player.highScore);pceFontPrintf(szBuffer)
;

    pceLCDPaint(0,108,70,8,8);
    BltGraphics(108,70,&pBmpChara,Player.c*8,0,8,8);    //    プレイヤーの色を表示
}
//----------------------------------------------------------------------
//    メインループ
```

★最近，自分のパソコン（Windows98）の起動に10分くらいかかり，USBもまったく認識しません。これはもしや…。（群馬県桐生市・かの20歳）……【**影：**そういう場合は，他人のパソコンを使う，これ，どう？**編：**影さん，私のパソコンで何やってるんですか！**影：**え？】

```
//------------------------------------------------------------------------
void MainLoop()
{
    switch(gameState)
    {
        case 0:GameInit();break;
        case 1:GameMain();break;
        case 2:GameOver();break;
        case 5:GameTitle();break;
    }
}
//------------------------------------------------------------------------
//    P/ECEの初期設定
//------------------------------------------------------------------------
void pceAppInit(void)
{
    unsigned char *vram;
    PBMP_FILEHEADER    pbhead;
    PCETIME    strTime;
    FILEACC fa;

    pceLCDDispStop();
    pceLCDSetBuffer(vbuff);
    pceAppSetProcPeriod(72);
    memset(vbuff,0,128*88);

    //    乱数初期化
    pceTimeGet(&strTime);srand(strTime.s100);

    //    グラフィック読み込み。
    memcpy(&pbhead,CHARA,sizeof(PBMP_FILEHEADER));
    pBmpChara.header=pbhead;
    pBmpChara.buf =CHARA+sizeof(PBMP_FILEHEADER);

pBmpChara.mask=CHARA+sizeof(PBMP_FILEHEADER)+(pBmpChara.header.h*pBmpChara.header.w)/4
;

    //    ハイスコアファイルの読み込み
    if (pceFileOpen(&fa,"umlaut.dat",FOMD_RD)==1){
        Player.highScore=500;    //    ファイルがないときは読み込みません
    }else{
        pceFileReadSct(&fa,&Player.highScore,0,sizeof(int));    //    データ読み込み
    }
    pceFileClose(&fa);

    //    グローバル変数初期化
    gameState=5;
    saveCheck=FALSE;

    //    その他初期化
    InitMusic();
    pceLCDDispStart();
}
//------------------------------------------------------------------------
//    アプリケーション・プロシージャ
//------------------------------------------------------------------------
void pceAppProc(int cnt)
{
    MainLoop();
    pceLCDTrans();
}
//------------------------------------------------------------------------
//    終了処理
//------------------------------------------------------------------------
void pceAppExit(void)
{
}
```

《リスト２》chara.cのプログラム・リスト

```
//バイナリデータ[chara.pgd]
//SIZE [164 byte]

const unsigned char    CHARA[] = {
    80,  77,  66,  80, 164,   0,   0,   0,   2,   1,  48,   0,   8,   0,   0,   0,
   144,   0,   0,   0,  85,  85,  15, 240,  14, 224,  10, 160,  15, 240,  15, 240,
   106, 171,  63, 252,  46, 236,  42, 168,  54, 108,  48,  12, 106, 171, 255, 255,
   238, 238, 170, 170, 217, 155, 192,   3, 106, 171, 255, 255, 238, 238, 170, 170,
   230, 103, 192,   3, 106, 171, 255, 255, 238, 238, 170, 170, 217, 155, 192,   3,
   106, 171, 255, 239, 238, 222, 170, 154, 230,  99, 192,  19, 106, 171,  63, 188,
    46, 108,  42, 104,  57, 140,  48,  76, 255, 255,  15, 240,  14, 224,  10, 160,
    15, 240,  15, 240, 255,  60,  60,  60,  60,  60, 255, 126, 126, 126, 126,  66,
   255, 255, 255, 255, 255, 129, 255, 255, 255, 255, 255, 129, 255, 255, 255, 255,
   255, 129, 255, 255, 255, 255, 255, 133, 255, 126, 126, 126, 126,  74, 255,  60,
    60,  60,  60,  60
};
```

iアプリ用



ついに舞台は携帯へ

とりでの攻防（こうぼう）

Shng（23歳）

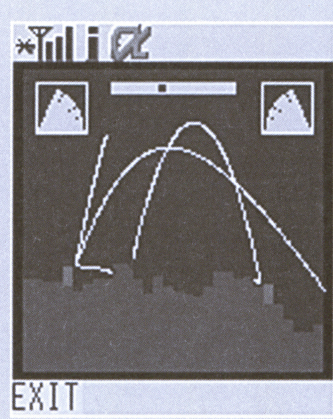
作品の内容

移植ものの定番『とりでの攻防』をiアプリに移植しました。

風の影響を考えながら，角度を決めて弾を発射，相手のとりでを先に壊した方が勝ちというゲームです。



《写真１》ｉアプリで，あの名作が遊べる



《写真２》正確な弾の発射が試される



《写真３》相手のとりでを破壊できるか？



《写真４》エミュレータでも実行ＯＫ。ただし速度が速すぎるかも？

操作方法

タイトルが表示されるので(**写真１**)，何か押せばゲームスタートです。ランダムで地形が生成されます。ゲームが始まると，プレイヤー側の攻撃から始まります。

中央上に風の強さが表示されています。左上に弾の発射角が表示されているので，ここだと思った角度で，何かキー押してください。弾があたると爆風で地形が壊れるので，それをうまく利用して相手のとりでを壊してください。爆風はだんだん大きくなってきます。

改造のアドバイス

・音をつける
・爆発のときに振動する
・速度が遅いと思った方は，wt関数をいじってください

エミュレータでの実行

ｉアプリ開発ツール(DoJaなど)で使用するエミュレータでも，基本的には問題なく実行できます。ただし，PCの動作速度によっては，角度設定の動きなどが速すぎてしまう場合があります。

苦労した点

ｉアプリでは，画面に描いた点が調べられなかったので，仮想画面を作って処理しました。本当は爆風を円形にしたかったのですが，爆風が小さいと四角に見えてしまい面白くなかったので，ひし形

★読むべし。よく読むべし。ひたすら読むべし。毎月読むべし。必ず読むべし。糸を引かなくなるまで読むべし。（群馬県高崎市・巽洋平23歳）……【影：キミは，カエルのべしか！編：解説しますと，べしは「〜だべし」，ケムンパスは「〜でやんす」としゃべる。】

にしてみました。

また，小数が使えなかったので，1024倍した値で計算しました。

★参考にしたもの

・1993年6月号 PC-98版「砦の攻防」(画面を参考にさせていただきました)

CHECKER FLAG

編：ベーマガでの投稿プログラムといえば，やはりこの作品ですね。元祖のPC88版から，最後にｉアプリ版まで掲載することができました。

つぐ美：だけど『とりでの攻防』には，やはり対戦モードがほしかったわね。携帯を交換しながらプレイする形式であっても，対戦モードを付けてほしかったな。

Dr.D：sinテーブルを1024倍という値で設定しておるのは，ビットシフト演算を用いるためじゃな。sin，cosで使用するときは，この値を右へ10ビットシフト(>>10)させると，1024で割った値と同じになる。このシフト演算は，通常の掛け算や割り算よりも処理速度が速いというメリットがあるのじゃ。

《リスト１》toride.javaのプログラム・リスト

```
// とりでの攻防

import com.nttdocomo.ui.*;
import java.util.Random;

public class toride extends IApplication {
    public void start() {
        TT t = new TT();
        Display.setCurrent(t);
        t.start();
    }
}

class TT extends Canvas{

int WIDTH = getWidth();
int HEIGHT = getHeight();
Graphics g = getGraphics();
Random rn = new Random();

// サインコサインテーブル sin[x], cos[90-x]  : 0<=x<=90
int sincos[] = {
      0,   17,   35,   53,   71,   89,  107,  124,  142,  160,  177,  195,
    212,  230,  247,  265,  282,  299,  316,  333,  350,  366,  383,  400,
    416,  432,  448,  464,  480,  496,  511,  527,  542,  557,  572,  587,
    601,  616,  630,  644,  658,  671,  685,  698,  711,  724,  736,  748,
    760,  772,  784,  795,  806,  817,  828,  838,  848,  858,  868,  877,
    886,  895,  904,  912,  920,  928,  935,  942,  949,  955,  962,  968,
    973,  979,  984,  989,  993,  997, 1001, 1005, 1008, 1011, 1014, 1016,
   1018, 1020, 1021, 1022, 1023, 1023, 1024 };
// 仮想画面
int STAGE[] = new int[120*120];

int pl;          // どちらの攻撃か
int r;           // 発射台の角度 (0~90)
int cr;          // コンピュータの選んだ角度
int x, y;        // 砲弾の座標
int vx, vy;      // 砲弾の移動量
int vvx;         // 風の影響
int w;           // 風の強さ
int ty1, ty2;    // とりでのＹ座標
int bo;          // 爆風の大きさ

int ox, oy;      // 原点座標
int wait;        // ウェイト


TT() {
    long l;
    wait = 0;
    l = System.currentTimeMillis() + 50;
    while (System.currentTimeMillis() < l) wait++;
    setSoftLabel(SOFT_KEY_1, "EXIT");
    ox = (WIDTH - 120)/2;
    oy = (HEIGHT - 120)/2;
}

public void start() {
    int i, j;
    for (;;) {
        // タイトル画面
        g.lock();
        g.setColor(g.getColorOfName(Graphics.BLACK));
        g.fillRect(0, 0, WIDTH, HEIGHT);
        g.setColor(g.getColorOfName(Graphics.BLUE));
        g.drawString("とりでの攻防", 20, 50);
        g.setColor(g.getColorOfName(Graphics.WHITE));
        g.drawString("とりでの攻防", 19, 49);
        g.drawString("HIT ANY KEY", 24, 80);
        g.unlock(true);
        while (!getKey());
        while (getKey());
```

```
// 画面作成
g.lock();
g.setColor(g.getColorOfName(Graphics.BLACK));
g.fillRect(0, 0, WIDTH, HEIGHT);
g.setColor(g.getColorOfName(Graphics.BLUE));
g.fillRect(ox, oy, 120, 120);
g.setColor(g.getColorOfName(Graphics.GREEN));
for (i=0; i<120; i++)
    for (j=0; j<120; j++)
        STAGE[i*120+j] = 0;
// 地形を作り，描画する
y = 90 + rn.nextInt() % 16;
for (x=0; x<30; x++) {
    for (i=y; i<120; i++)
        for (j=0; j<4; j++)
            STAGE[i*120+x*4+j] = 1;
    g.fillRect(ox+x*4, oy+y, 4, 120-y);
    // とりでを作る
    if (x == 4 || x == 24)
        for (i=1; i<9; i++)
            for (j=0; j<4; j++) {
                STAGE[120*(y-i)+x*4+j] = 2;
        if (x == 4)
            ty1 = y - 8;
        else if (x == 24)
            ty2 = y - 8;
    }
    y += rn.nextInt() % 8;
    if (y > 110) y = 110;
    if (y < 60) y = 60;
}
// とりでを描く
g.setColor(g.getColorOfName(Graphics.GRAY));
g.fillRect(ox+16, oy+ty1, 4, 8);
g.fillRect(ox+96, oy+ty2, 4, 8);
pl = 0;
bo = 4;
g.unlock(true);
// メインループ
for (;;) {
    w = rn.nextInt() % 5;
    r = 91;
    // 風の強さを描く
    g.lock();
    g.setColor(g.getColorOfName(Graphics.AQUA));
```

```
    g.fillRect(ox+35, oy+5, 50, 5);
    g.setColor(g.getColorOfName(Graphics.BLUE));
    g.fillRect(ox+59+w*5, oy+6, 3, 3);
    g.unlock(true);
    // プレーヤーの発射角度を決める
    if (pl == 0) {
        for (;;) {
            g.lock();
            // 90度を越えたら枠クリア
            if (r > 90) {
                g.setColor(g.getColorOfName(Graphics.BLUE));
                g.fillRect(ox+6, oy+6, 18, 18);
                g.setColor(g.getColorOfName(Graphics.WHITE));
                g.drawRect(ox+5, oy+5, 20, 20);
                r = 0;
            }
            g.setColor(g.getColorOfName(Graphics.AQUA));
            g.drawLine(ox+6, oy+24, ox+6+((sincos[90-r]*18)>>10), oy+24-
((sincos[r]*18)>>10));
            // キーが押されたら発射の準備
            if (getKey()) {
                x = 21 << 10;
                y = ty1 << 10;
                vx = sincos[90-r] >> 1;
                vy = -sincos[r]-10;
                vvx = 0;
                break;
            }
            r++;
            g.unlock(true);
            wt();
        }
    } else {
        // コンピュータの発射角度を決める
        getKey();
        cr = Math.abs(rn.nextInt() % 91);
        for (;;) {
            g.lock();
            // 90度を越えたら枠クリア
            if (r > 90) {
                g.setColor(g.getColorOfName(Graphics.BLUE));
                g.fillRect(ox+96, oy+6, 18, 18);
                g.setColor(g.getColorOfName(Graphics.WHITE));
                g.drawRect(ox+95, oy+5, 20, 20);
                r = 0;
```

★RGBとRPGをいつも間違うので何とかしてください。（岩手県釜石市・**鉄人坂井25歳**）……【**影：**640×480ドット，16色表示が可能な標準の画面サイズをなんていったっけ？AVG？**編：**それはアドベンチャーゲーム。**影：**あ，VGAか。横文字はややこしい。】

```
            }
            g.setColor(g.getColorOfName(Graphics.AQUA));
            g.drawLine(ox+114, oy+24, ox+114-((sincos[90-r]*18)>>10), oy+24-
((sincos[r]*18)>>10));
            // コンピュータが決めた発射角度になったら発射の準備
            if (r == cr) {
                x = 95 << 10;
                y = ty2 << 10;
                vx = -(sincos[90-r] >>1);
                vy = -sincos[r]-10;
                vvx = 0;
                break;
            }
            r++;
            g.unlock(true);
            wt();
        }
    }

    g.setColor(g.getColorOfName(Graphics.WHITE));
    // 砲弾が飛ぶ
    for (;;) {
        getKey();
        x += vx;
        y += vy;
        vx += (vvx >> 7);
        vy += 7;
        vvx += w;
        g.lock();
        g.fillRect(ox+(x>>10), oy+(y>>10), 1, 1);
        g.unlock(true);
        // 弾が画面外に出たら次の処理に移る
        if ((x>>10) < 0 || (x>>10) > 119 || (y>>10) > 119)
            break;
        if (y > 0) {
            // 着弾判定
            if (STAGE[(y>>10)*120+(x>>10)] != 0) {
                zeroEXP(x>>10, y>>10, bo>>1);
                g.setColor(g.getColorOfName(Graphics.RED));
                for (i=0; i<(bo>>1); i++) {
                    g.lock();
                    fillEXP(ox+(x>>10), oy+(y>>10), i);
                    g.unlock(true);
                    wt();
                }
                g.setColor(g.getColorOfName(Graphics.BLUE));
                for (i=0; i<(bo>>1); i++) {
                    g.lock();
                    fillEXP(ox+(x>>10), oy+(y>>10), i);
                    g.unlock(true);
                    wt();
                }
                break;
            }
        }
        wt();
    }
    bo++;
    if (bo > 40)
        bo = 40;
    pl = pl == 0 ? 1 : 0;
    // プレーヤーのとりでをチェック
    if (STAGE[ty1*120+18] * STAGE[ty1*120+19] * STAGE[(ty1+1)*120+18] *
STAGE[(ty1+1)*120+19] == 0) {
        // ゲームオーバー処理 (プレーヤーの負け)
        g.setColor(g.getColorOfName(Graphics.RED));
        for (i=0; i<30; i++) {
            g.lock();
            fillEXP(18+ox, ty1+4+oy, i);
            g.unlock(true);
            wt();
        }
        g.setColor(g.getColorOfName(Graphics.BLUE));
        for (i=0; i<30; i++) {
            g.lock();
            fillEXP(18+ox, ty1+4+oy, i);
            g.unlock(true);
            wt();
        }
        g.lock();
        g.setColor(g.getColorOfName(Graphics.BLACK));
        g.fillRect(ox+33, oy+50, 50, 16);
        g.setColor(g.getColorOfName(Graphics.WHITE));
        g.drawString("YOU LOSE", ox+35, oy+61);
        g.unlock(true);
        // キー入力待ち
        while (getKey()) ;
        while (!getKey()) ;
        break;
    } else
```

★こんにちは！ご無沙汰してます。突然ですが，3月号に僕が載ってました。19ページの高専ロボコンのところなんですが，僕は北九州高専のFlexメンバーとして出場してました。なんか，うれしかったです。（福岡県宗像市・**Shu17歳**）……**【編：どれ，胸に人形を？】**

```
            if (STAGE[ty2*120+96] * STAGE[ty2*120+97] * STAGE[(ty2+1)*120+96] *
STAGE[(ty2+1)*120+97] == 0) {
                // ゲームオーバー処理（プレーヤーの勝ち）
                g.setColor(g.getColorOfName(Graphics.RED));
                for (i=0; i<30; i++) {
                    g.lock();
                    fillEXP(98+ox, ty2+4+oy, i);
                    g.unlock(true);
                    wt();
                }
                g.setColor(g.getColorOfName(Graphics.BLUE));
                for (i=0; i<30; i++) {
                    g.lock();
                    fillEXP(98+ox, ty2+4+oy, i);
                    g.unlock(true);
                    wt();
                }
                g.lock();
                g.setColor(g.getColorOfName(Graphics.BLACK));
                g.fillRect(ox+33, oy+50, 50, 16);
                g.setColor(g.getColorOfName(Graphics.WHITE));
                g.drawString("YOU WIN!", ox+35, oy+61);
                g.unlock(true);
                // キー入力待ち
                while (getKey()) ;
                while (!getKey()) ;
                break;
            }
        }
    }
}

// キーの状態を調べる＆終了ボタンチェック
public boolean getKey() {
    int k;
    k = getKeypadState();
    if ((k&(1<<Display.KEY_SOFT1)) != 0)
        IApplication.getCurrentApp().terminate();
    if (k != 0)
        return true;
    else
        return false;
}

// 爆風の描画
public void fillEXP(int cx, int cy, int cr) {
    int i;
    cy -= cr;
    for (i=cr; i>=0; i--) {
        g.fillRect(cx+i, cy+i, 1, (cr-i)*2);
        g.fillRect(cx-i, cy+i, 1, (cr-i)*2);
    }
}

// 爆風の描画 仮想画面用
public void zeroEXP(int cx, int cy, int cr) {
    int i, j;
    cy -= cr;
    for (i=cr; i>=0; i--) {
        for (j=0; j<(cr-i)*2; j++) {
            if (cx-i > 0 && cx-i < 120 && cy+i+j > 0 && cy+i+j < 120)
                STAGE[(cy+i+j)*120+cx-i] = 0;
            if (cx+i > 0 && cx+i < 120 && cy+i+j > 0 && cy+i+j < 120)
                STAGE[(cy+i+j)*120+cx+i] = 0;
        }
    }
}

// ウェイトをとる
public void wt() {
    int i;
    for (i=0; i<wait; i++) {}
}

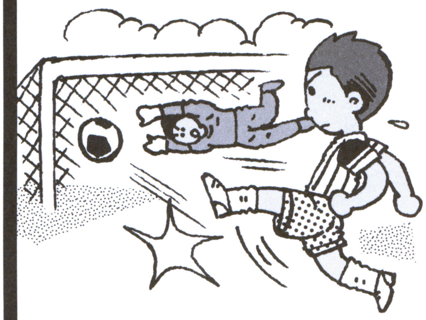
public void paint(Graphics g) {}

}
```

《リスト２》toride.jamのプログラム・リスト

```
PackageURL = toride.jar
AppSize = 3268
AppName = toride
AppVer =
AppParam =
AppClass = toride
SPsize =
LastModified = Mon, 06 Jan 2003 12:20:37
KvmVer =
```

★プログラム投稿のページで“参考にした本”が気になります。（石川県羽咋郡・**メーテルとリンク34歳**）……【**編**：どんな本を参考にしているかはやっぱり気になりますよね。ベーマガは，プログラムリストを読んで，どんな組み方をしているか勉強できますよ。】



遊び要素の入ったPK

AFO（エーエフオー）のPK（ピーケー）

おぐりん　(25歳)

はじめに

これは，私がベーマガに投稿したｉアプリの第３作目です。

といっても，このペンネームで投稿するのは初めてなので，私が誰なのかわからないでしょうね。このゲームは，８年前に私の友人「おぐりん」(ちなみに，「おぐりん」は当時呼んでいたニックネームではありません。彼の本名で投稿するわけにはいかないので，わざわざ作ったものです）と共同して制作し，ベーマガに投稿したプログラムなので，彼に敬意を表して，この名前を使うことにしました。

まあ，しばらく読んでいけば私が誰なのかわかると思います。

内容・遊び方

このゲームは，ワンキーのサッカーゲームです。もっともサッカーゲームとはいっても，ペナルティ・キックしかできません。個人的には，チープなグラフィックと奇妙な操作性から，「バカゲー」と思っているのですが，どうでしょう。

ゲーム中は，赤色の人を動かしてPKをします。青色の人は，コンピュータが操作します。

タイトル画面(**写真１**)で三つのモードが選択でき，それぞれ以下のような内容になっています。

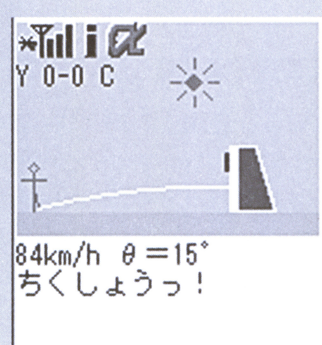
シュート練習の場合には，操作キーを２回押して，ひたすらシュートを決めてシュート成功率の高さを競います。

キーパーの練習の場合には，シュートの方向にうまくジャンプさせてシュート成功率を低さを競います。

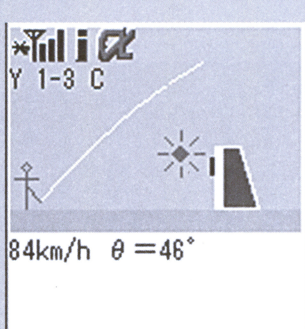
「COMと勝負」は，５回勝負のPK戦です。ゴールを多く決めた方の勝ちですが，同点の場合は，決着がつくまで延長戦を行います。



《写真１》おバカ要素満載(？)のPKゲーム



《写真２》華麗なシュートを決めろ！



《写真３》何と太陽が落下する!?この後，面白い画面に？



《写真４》開発用エミュレータでもバッチリ遊べます

操作方法

各モード共通で，ソフトキー１は時間調整，ソフトキー２はゲーム中断・終了に使います。その他のキー(ここでは操作キーと呼ぶことにします)は，すべて同じ機能を果たします。

もっと詳しく説明すれば，シュート練習の場合には，操作キーでシュートの速度・角度の順に設定します。キーパーの練習の場合には，操作キーを押してキーパーをジャンプさせます(もう一度操作キーを押すとキーパーが降下します)。

ゴールなどの判定

ゴールされたかどうかは，ボールの軌跡がゴールポストの左端を通過したかど

★昔のパソコンゲーム（5インチ）がすべて起動しなくなった…。さらば！我が青春！（神奈川県平塚市・**エメドラやりてぇ～24歳**）
……【**影**：逆を言うと，よく今の今まで起動していたと言えるんじゃないか。**Dr.D**：ワシの家には8インチのフロッピーが！】

《第１表》変数表

boolean sk1, sk2, kyin	ソフトキー１・２，それ以外が押されたか
int gkx, gky	キーパーの座標
int snx, sny	太陽の座標
int wt	足の動きのウェイト
int bdg, vbdg, pw	足の角度・速度，シュートの強さ
int bx, by	ボールの座標
int bvx, bvy	ボールの速度
int gkvy	キーパーの速度
int tryn[], gl[]	シュート回数，ゴール回数
String rslt[]	シュートの結果
int mcos[]	1000×cos(0°～90°)

うかで判定します。したがって，そこを通過せずに軌跡がゴールの中に入ってもゴールしたと判定されません。その判定の仕方は，ボールの軌跡を描いている線分が，ゴールポストの左端の線分と交差しているかどうかで判断します(関数flgoal()を参照)。

キーパーがキャッチしたかどうかも同様に，キーパーの前方のみで判定します。

シュート結果のコメント

シュート結果の判定は次の四つのコメントと記号でわかります。

・シュートが決まったとき…「ちくしょうっ！」(サッカーボール)

・シュートをはずしたとき…「お見事っ！」(斜め矢印，汗)

・キーパーが受けたとき …「大当たりー！」(じゃんけんのパー)

他にも，ある条件を満たしたときのコメントがあります。

プログラムについて

このプログラムの目的は，不動小数点計算のできないｉアプリで三角関数を使うことにあります。2003年１月号掲載の『幻影転移』の説明にあるように，三角関数を使う場合，多くの改造が必要になり，その作成に時間がかかってしまいました。

ｉアプリで三角関数を使う場合は，0°～90°のコサインを何倍かにして配列で用意(このプログラムでは1000倍にしてmcos[]に代入している)すればできます。それを基にして，あらゆる角度のサイン，コサインを求めることができます(関数getmsin()，getmcos()を参照)。

今回のプログラムのもう一つの目的は，何回も繰り返す命令をメソッドにしてつくることにあります。ｉアプリの場合は，実行ファイルのサイズの制限があるので，メソッドを少なくする必要があると言われています。

オブジェクト指向でプログラム設計をすると，わかりやすいプログラムになると一般的には言われています(ベーマガ2003年２月号113ページ参照)が，無理にオブジェクト指向でプログラムして，各オブジェクトの関係がわからなくなってしまうと，逆にわかりにくくなるような気がします。

エミュレータでの動作

エミュレータ(J2ME Wireless SDK for DoJaなど)を使って遊ぶ場合(**写真4**)，音楽が再生されません。『幻影転移』では，音楽再生部分の例外処理をつけ忘れていたので，エラーが発生しました。しかし，このプログラムには，例外処理を施してあり，その部分をとばして処理するので，エラーにはなりません。

ＡＦＯの由来

詳しく書くことはできませんが，「ＡＦＯ」は，中学・高校の同級生の名字が由来です(「ＡＦＯ」は「エー・エフ・オー」と読みます。「あふぉ」と読んでもらっても構いませんが…)。もちろん，その人に向かって「ＡＦＯ」と言ったことはありません(もっとも，面と向かって言っても怒るような人ではないですが……)。

８年も経つといろいろと変わるもので，「ＡＦＯ」は結婚し，「おぐりん」は，交通事故で帰らぬ人となりました。

そのままだと全くおもしろくないプログラムが「おぐりん」の一言で，もう一歩で掲載できるレベルのプログラムにまで化けたという記憶が，今でもよみがえってきます。

Pick up LIST

《リスト》三角関数関連のリスト

```
// cos値×1000の値の定義
static final int mcos[]={ //1000×cos (0° ～90° )
    1000,1000,999 ,999 ,998 ,996 ,995 ,993 ,990 ,988 ,
    985 ,982 ,978 ,974 ,970 ,966 ,961 ,956 ,951 ,946 ,
    940 ,934 ,927 ,921 ,914 ,906 ,899 ,891 ,883 ,875 ,
    866 ,857 ,848 ,839 ,829 ,819 ,809 ,799 ,788 ,777 ,
    766 ,755 ,743 ,731 ,719 ,707 ,695 ,682 ,669 ,656 ,
    643 ,629 ,616 ,602 ,588 ,574 ,559 ,545 ,530 ,515 ,
    500 ,485 ,469 ,454 ,438 ,423 ,407 ,391 ,375 ,358 ,
    342 ,326 ,309 ,292 ,276 ,259 ,242 ,225 ,208 ,191 ,
    174 ,156 ,139 ,122 ,105 ,87  ,70  ,52  ,35  ,17  ,0
};

/* コサインを求める */
public int getmcos(int deg){
    int c=(deg/90)%4; int d=Math.abs(deg%90);
    switch(Math.abs((deg/90)%4)){
        case 0:
            c=mcos[d];
            break;
        case 1:
            c=-mcos[90-d];
            break;
        case 2:
            c=-mcos[d];
            break;
        case 3:
            c=mcos[90-d];
            break;
    };
    return c;
}

/* サインを求める */
public int getmsin(int deg){
    return getmcos(90-deg);
}
```

CHECKER FLAG

編：ワンキーで遊べるPKゲームですが，シュートの力加減と角度の調整が意外と奥深く，なかなか遊べる作品です。

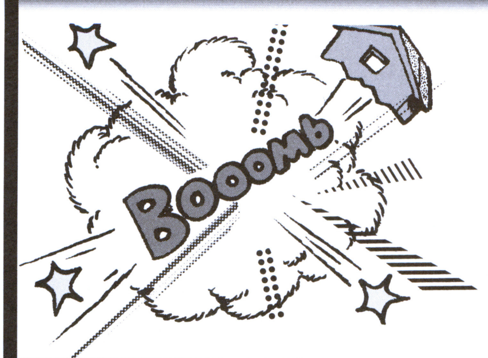
Dr.D：最近では見なくなったが，市販されたサッカーゲームでも，PK戦は左右真ん中のどこかにシュートするという，いい加減な内容のものもあった。その点を考えると，横視点固定ではあるが，このPKゲームは完成度が高いと言えるじゃろう。

影：全体的にバカゲーっぽい雰囲気を作り出している点に好感がもてるなぁ。

つぐ美：太陽に向かってボールを蹴ると…何と，写真３のような事態に！

★巷ではブレイクダンスが流行っているみたいですよ(液キ○べが踊る理由)。(茨城県稲敷郡・**雪崩式21歳**)……【**影：**よっ，よっ，よっと。**つぐ美：**へんな格好！**影：**マイケル・ジャクソンのムーンウォークだよ。これも，ブレイクダンスの一つなんだな。よっよっと。】

PC-9801(N88-DiskBASIC)用



無敵状態で障害物を爆破

バクハゲーム

吉村　直繁（24歳）

作品の内容

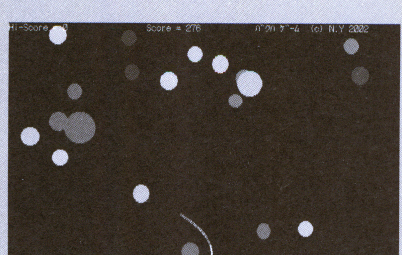
一画面プログラムでワンキーゲームです(**写真1**)。お手軽ですね。丸い円(障害物)にぶつかるとゲームオーバー。左右にかわしながら前に進んでください。

おっと，もちろんそれだけではありません。黄色い円にはぶつかっても大丈夫です。ある程度の時間，自機の色が変化し無敵になれます。得点も入ります。

無敵になると，障害物を体当たりで爆破でき，得点になります。ただし，一度爆破すると無敵ではなくなり，自機は白色に戻ります。

また，連続で黄色い円に当たりつづけると，無敵時間は蓄積されていきます。爆破で入る得点は，そのとき無敵時間がいくら蓄積されているかに比例します。

そうそう，爆破による得点は，自機下に表示されます。では，ガンガン黄色にぶつかり，どんどん爆破して高得点をめざしてください。



《写真1》状態しだいで，障害物を破壊できるのが特徴

操作方法

スペースキーを押している間は右に加速，離すと左に加速します。ゲームオーバーになったらリターンキーで再びゲームスタートです。

《第1表》変数表

X,Y	自機座標
SC#	スコア
HI#	ハイスコア
D#	進んだ距離
V	自機の横方向の速度
MT	無敵時間
WA,WB	ウェイト
LT	1回の爆破か判定する用のタイマー
PT#	1回の爆破による得点
N	円の半径に関係する値
P	自機座標の色

CHECKER FLAG

編：ベーマガらしく，最終号でもN88-BASICの投稿作品を掲載してみました。

つぐ美：障害物を破壊できるのが面白いですね。ただ，黄色い円にぶつかると無敵になるのですが，その自機の無敵状態の表示が，チョットわかりにくいですね。

影：単純なゲーム内容なのだから，自機状態も，見た目でパッと理解できる工夫がほしかったところだな。

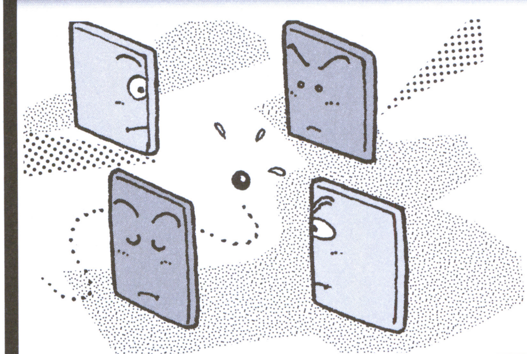
Dr.D：もう一つの特徴である短く作られたリストは，やはりキチンと評価したいものじゃな。当たり判定を，基本的にPOINT命令だけで行っておるのが，短さの要因じゃ。ただ，短いリストでも，最低限のコメント記述は行ってほしかったのぉ。

《リスト》バクハゲームのプログラム・リスト

```
10 SCREEN 0:WIDTH 80,25:OUT &H68,6:CONSOLE 0,25,0,1
20 CLS 3:COLOR 7,,,,2:LOCATE 50,0:PRINT "ﾊﾞｸﾊ ｹﾞｰﾑ (c) N.Y 2002"
30 Y=155:X=320:SC#=0:D#=0:V=0:MT=0:WA=200:WB=900:RANDOMIZE VAL(RIGHT$(TIME$,2))
40 LOCATE 0,0:PRINT "Hi-Score =";HI#
50 ROLL -1:A=(RND<.1)*10
60 IF RND<.09 THEN CIRCLE(RND*639,7+N-A),RND*5+10+N-A,RND*5+1,,,,F
70 P=POINT(X,Y-1):LT=LT+(LT>0)
80 CIRCLE(X,Y),1,6-(MT<80)-(MT/60 MOD 6):IF P THEN *OVER
90 V=V-.1-(INP(&HE9)=191)*.2:X=X+V:MT=MT+(MT>0):X=X+(X>639)*639-(X<0)*639
100 SC#=SC#+1:D#=D#+1:LOCATE 28,0:PRINT "Score =";SC#
110 N=INT(D#/500):FOR W=0 TO WB:NEXT:GOTO 50
120 *OVER:A=-(P<>6 AND MT=0)*100+100:BEEP 1:FOR J=0 TO 1:FOR I=0 TO A STEP 2
130 CIRCLE(X,Y),I,2+(J=1)*2:FOR W=0 TO WA:NEXT:NEXT:NEXT:BEEP 0
140 IF P=6 THEN MT=MT+150:SC#=SC#+MT:PT#=MT:GOSUB *PT:GOTO 90
150 IF MT>0 THEN PT#=(MT+N*10+5)*10:SC#=SC#+PT#:MT=15+N:GOSUB *PT:GOTO 90
160 FOR I=0 TO 500:PSET(RND*639,RND*199),RND*7:NEXT
170 LINE(26*8,8*8-1)-(48*8,13*8),1,BF,9
180 COLOR 2:LOCATE 27,8:PRINT "**** Game Over ****"
190 IF SC#>HI# THEN COLOR 6:LOCATE 27,10:PRINT "You Get Hi-Score !!":HI#=SC#
200 COLOR 4:LOCATE 27,12:PRINT "** Hit Return Key **"
210 IF INKEY$=CHR$(13) THEN 20 ELSE 210
220 *PT:LOCATE 0,22:PRINT SPC(79):XX=X/8-4:XX=XX+XX*(XX<0)+(XX-69)*(XX>69)
230 PT#=PT#-P#*(LT>0):LOCATE XX,22:PRINT PT#"pts":P#=PT#:LT=N+15:RETURN
```

★川上氏，MML講座どうでしょう？／試験前にノートPCが壊。BIOS反応なし(涙)…もしや。(熊本県玉名郡・**巨石人21歳**）……【**影**：私のPCがハングアップ。最近多いんだなー，これが。なんか不吉な予感。むむ，みんなの目が！ん？気まずい雰囲気。仕事しまーす。】

ポケコン(PC-G850)用



1ドットのキャラクタを操作

よけろ点(てん)

広岡　聡之（16歳）

はじめに

こんにちは。1行でできるプログラムを作りました。手軽なので，ぜひ入力してください。

入力・実行方法

PROGRAM MODEで間違いなくリストを入力したあと，RUN MODEでRUNと入力し，リターンキーを押すとプログラムが実行されます。モードについての詳細は，ポケコンの説明書を見てください。

実行したらすぐゲームが始まります。ゲームが終了したあと，再びプレイするには，もう一度実行してください。

ルールと操作方法

ポケコンで定番のドット避けゲームを，自分なりのルールにアレンジしたものです。実行すると画面上に縦線がたくさん出現します。そしてすぐに左から点が出てきます。それを画面右端まで無事に移動させてください。途中の縦線にぶつかるとゲームオーバーです。

点の動きは，以下のようになっています。

・横方向について：キーを押すと左に，放すと右の方向に動く

・縦方向について：画面上下にぶつかると逆方向に進む

動き方がよくわからない場合は，キーに触れずにやってみてください。そうするとわかると思います。

それと，点は動いているので，非常に見にくいと思います。点に集中していないと見失うことがあります。

そういう時は，画面コントラストをあげてみてください。あと，左上の数字はステージ数です。

最後に

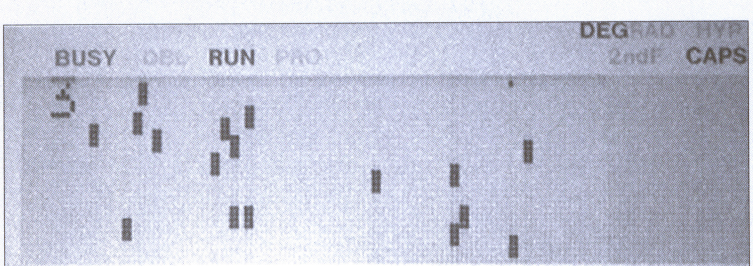
1行プログラムは自分のホームページでも公開しています。

マイホームページ：

http://www.nextftp.com/toshi_home/

《リスト》色暗号のプログラム・リスト

```
1 FOR L=1 TO 99:CLS :PRINT STR$ L:X=0:Y=23:F=1:FOR J=1 TO L*6:P= RND 99+Y:Q= RND 44:LINE (P,Q)-(P+1,Q+3),B:NEXT :REPEAT :PSET (X,Y):F=((Y=47)-(Y=0))*2+F:BEEP 1,,80:PRESET (X,Y):Y=Y+F:X=X-( INKEY$ ="")*2-1:UNTIL POINT (X,Y):IF X=144 NEXT
```



《写真1》ドットの進行方向を変えて障害を避けろ

CHECKER FLAG

Dr.D：残念なことに，ポケコン以外ではほとんど見かけなくなってしまった1ラインプログラム。ゲーム内容はスタンダードな障害物避けの内容じゃが，自動的に移動していく点の動きや，進行方向を切り替えていくという操作方法が，ゲームを面白く仕上げておる。

つぐ美：1ドットの点という，ホントに最小限の形で表現された自キャラでゲームを制作するってのも，自作プログラムならではの魅力ですよね。

★イラク戦争とパソ破戦争がこのまま続くと，今後ベーマガはどうなるのでしょう…(ーー;)。（福島県郡山市・近内沖尋20歳）……【編集長：ベーマガも今月号で休刊となってしまいました。世界の平和と，みなさんの活躍を願って…長い間ありがとうございました。】

★★★★★★★ マイコンBASICマガジン ★★★★★★★

明日のスター・プログラマー

Programmer's Room

今月の受賞作

毎月すべての掲載プログラムの中から，アイデア，リストの面で優れている作品を，それぞれ「ベストアイデア」「ベストリスト」，そして最も優れていると思われる作品を「ベストプログラム」として選考していきます。

今月の受賞作品と編集部からのコメントは右の表のとおりです。

ベストプログラム	PIECE用	ぷれす６０	ていこく(23歳)
コメント：すっきりとまとめられたプログラムと，簡単なゲームシステムのおかげで誰もが白熱できる点を評価して，今月のベストプログラム受賞です。(81ページ掲載)			
ベストアイデア	VB.NET用	ティ取	松江　祐輔（21歳）
コメント：ティッシュペーパーをひたすらとり続けるという，意外なものをゲーム化したという点を評価して，ベストアイデア受賞です。(60ページ掲載)			
ベストリスト	HSP用	LineBall	稲津　和磨（17歳）
コメント：ゲーム内容や演出効果に対してリストの長さが短めなので，この点を評価してのベストリスト受賞です。(75ページ掲載)			

読者からのご意見

おぐりんさん（25歳）

今でもベーマガに投稿しようと思うのは，高校時代の記憶を懐かしんでいるのでしょう。

大学受験の勉強に追われながらも，ベストプログラムに選ばれる赤羊旅団さんのプログラムを見ると，とくに思い出されます。

稲津和磨さん（17歳）

やっと２作目の投稿です(前は没った)。今回のゲームは，弟のウケがよかったので投稿しました。普段からコメント不足だったので，付け足すのが大変！！これからは，しっかりコメント付けて書こーっと。

らぅ抹茶さん（23歳）

プログラムリストに対してのコメントの付け方などを記事にしてみてはどうでしょうか？コメントのあるリストと，ないリストでは見方もがらりと違うわけだし…。

編集部より

さまざまなご意見，ホントにうれしかったです。残念ながら今月号をもって休刊となってしまうのですが，こういった読者からの声に支えられて，ここまでベーマガを発行し続けることができました。

これからも，お仕事・ホビー・学業と，さまざまな分野での，皆さまのプログラマーとしてのご活躍をお祈りいたしております。

投稿ありがとう

よよい／のり／梶原寿樹／らぅ抹茶／寺内洋平／∞カロリー吉野家／赤羊旅団／橘　一馬／米久保　直樹／いしかね　みひろ／エレファント／笹原翔太／nalto／古川和広／Patrosystem／M.Hayashi／いわし／和々草／マスター／d金魚／石山明弘／ha55／伊藤優／あずき色／南とおれが／

※紹介できなかった方，そして掲載できなかった方ごめんなさい。

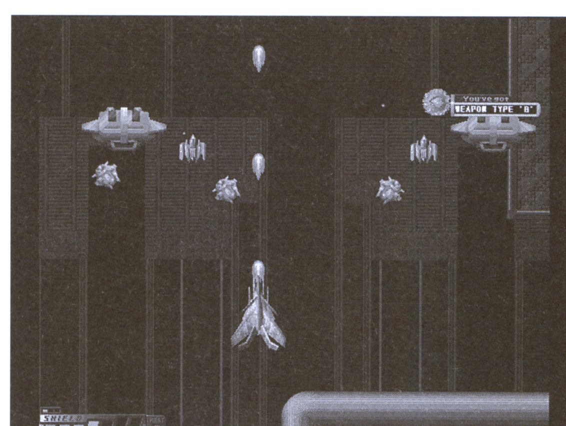
最後に，今までベーマガに投稿してきてくれたみんな，ほんとうにありがとう！！

ベーマガ名作劇場

このコーナーでは，今まで投稿プログラムコーナーに掲載された作品の中から，編集部が名作として選んだ作品を紹介していきます。

今月の名作：『CARDINAL PROJECT』(2001年３月号掲載作品 木むらさとし作)

○ノート



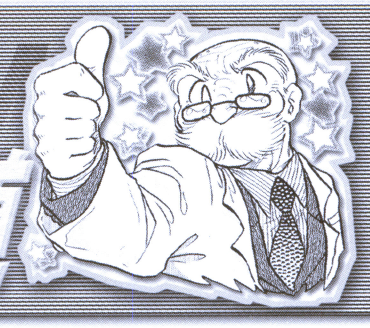
当時連載されていた企画「牛丼食えスト」のプログラム制作を担当していた木むらさとし氏が，VisualC++とDirectXを使用して制作した超大作投稿作品。

投稿プログラムで“超大作”というジャンルが設けられてから，完成度の高いプログラムが数多く投稿されてきたが，その中でもこの作品の完成度は群を抜いていた。

CHECKER FLAGでも絶賛のコメントを得たが，その中でDr.Dが「この調子で『牛食え』のほうも頑張ってほしいものじゃ」と述べた次の号から，その「牛丼食えスト」が休載となってしまったのは，何とも皮肉だった。

今月号で本誌が休刊となってしまうため，この作品が，投稿プログラムコーナーにおいて高い完成度を誇った，最後の超大作投稿プログラムと言えるだろう。

Dr.Dのベスト・プログラムへの道



今月のテーマ：『テキストを符号化しよう』

今月は，「テキストを符号化しよう」をテーマに，リスト内などに記述する日本語などのテキストデータを符号化(数値化)する方法を解説をしていこう。

これは，リスト掲載を主体としてきたベーマガにおいて，特にRPGなどを投稿する際には重宝されたテクニックじゃ。もう投稿で使用することはないじゃろうが，こういったテクニックをしっかりと身につけて，将来のベストプログラマーを目指してほしい。　**(Dr.D)**

符号化とは？

まず符号化とはなんぞや，ということから説明していこう。

符号化とは，簡単に言うと「文字を数値で表す」ということを示す。例えば「半角のAは，ASCIIコード65で示される」といった具合じゃな。

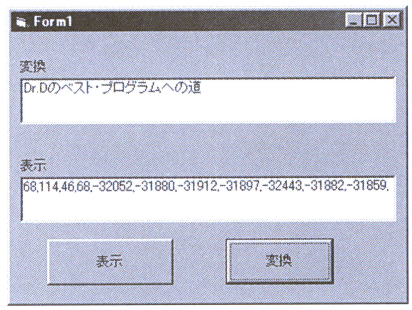
RPGなどの投稿作品では，文章データを符号化して数値で示すことで，リストから見ただけでは，どのような文章が定義されているかが，わからなくなるというわけじゃ。

VBで符号化を実現する

符号化の原理が理解できたところで，この符号化処理をVisualBasicで再現してみよう。

ここで紹介するプログラムは，**写真1**のようなものになり，上のテキストボックスに入力した文字列を符号化して，下の文字列へ表示するというものじゃ。

フォームの配置図などは，誌面の関係で割愛させてもらうが，このプログラムは**リスト1**のように形で作られておる。なお，プログラム制作の参考のために，文字の符号化だけでなく，符号



《写真1》符号化処理の例

化したデータが保存されたテキストファイルを読み込んで，再び文字として表示する機能も備えておる。また，生成したコードをテキストファイルに保存する処理も備えてある。詳しくはリスト内のコメントを合わせて参照してほしい。

さて，このプログラムで肝になるのは，Asc()命令とChr()命令じゃ。

Asc()命令の方は，引数に文字列を指定して，その文字列のコード(符号化されたもの)を返す。つまり，符号化する際に使用する命令というわけじゃ

逆にChr()命令は，引数に文字コードを指定することで，そのコードに対応する文字を返す。これは，符号化された文字を元に戻すときに使用することになる。

中には「日本語は2バイトで表現されているけど，そのまま指定して大丈夫なの？」と思う人もいるかもしれんが，Asc()／Chr()命令では，特別な処理を必要とせず，そのまま日本語(2バイト文字)を引数にしたり，返り値として受け取ることができる。

このプログラムを参考にすれば，読まれたくないデータを簡単に数値化(符号化)できるようになる。

こういったデータを扱う処理をマスターすることで，プログラムにおける守備範囲が広がる。ぜひ，ここまで解説してきた符号化をマスターしてほしいものじゃな。

最後に

本当は，この符号化処理を元に，暗号化やデータ圧縮などの解説も行っていこうと思っていたのじゃが，残念ながらベーマガ休刊に伴い，それはできなくなってしまった。

そこで，この暗号化やデータ圧縮については，このページを読んでくれた読者への(提出期限のない)宿題としておこう。残念ながらワシが採点することはないが，この暗号化やデータ圧縮を自分のプログラムに流用できたら，それがこの宿題の解答ということになるじゃろう。

今まで掲載・解説されてきたプログラム・記事を参考に，今のベーマガ読者が近い将来のコンピュータ・ゲーム業界を支えるプログラマーになることを願いつつ，このコーナーを終了させてもらうことにしよう。

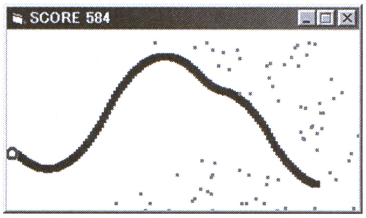
掲載プログラムを改造してみよう

このコーナーでは，毎月掲載されたプログラムの中から1～2本ピックアップして，そのプログラムの改造を紹介していくゾ。

VisualBasic用『点つぶし』を改造しよう

●改造「触手の長さを変更しよう」

このプログラムは，ゲームで使用する主なキャラクタの値設定を，リスト内の定数定義(**リスト2**)で行っているため，この定数の値を書き換えることで，触手の長さをはじめ，敵の数なども変更が可能なのじゃ(**写真2**)。



《写真2》『点つぶし』の改造の結果

3月号の改造でも触れたが，一歩進んだプログラマーを目指すのであれば，このような定数定義の記述を行うクセを身につけてほしいものじゃな。

iアプリ用『とりでの攻防』を改造しよう
●改造「プレイヤーとの対戦を実現させよう」

CHECKER FLAGでつぐ美クンが述べている対戦モードを実現させるための改造じゃ。

改造方法としては，COM処理として記述されている部分もプレイヤーが操作できるようにするというものじゃ。

リスト3のような改造を行うことで対戦モードが実現できる。ただし，この対戦モードは，一つの携帯電話で遊ぶ形式のため，プレイヤーの順番が代わるたびに携帯電話自体を交換しながら遊ぶことになる。

この改造記述を参考に，対COM戦と対プレイヤー戦をタイトルで選択できるようにしてもよいじゃろう。

《リスト1》VBでの符号化プログラムのリスト

```
'********************************
' VBでの文字列符号化プログラム
' 制作：Dr.D
'********************************

' --- グローバル変数の定義 ---
'変換可能な最大文字数
Const MAX = 100

'****************************
' Text1内の文字列を符号化する
'****************************
Private Sub Command2_Click()
    Dim tmp As String
    Dim length, i As Long
    Dim num(MAX) As Long

    '下のテキストボックスを初期化
    Text2 = ""

    '文字列＆文字列の長さを取得
    moji = Text1: length = Len(moji)

    '文字列の長さが，定義している最大数(MAX=100)以下かチェック
    If length >= MAX Then
        MsgBox "範囲をオーバーしました"
        End
    End If

    '文字列を符号化
    For i = 1 To length
        tmp = Mid(moji, i, 1)
        num(i - 1) = Asc(tmp)
    Next i

    '符号化したデータを表示
    For i = 0 To MAX
        Text2 = Text2 & num(i)
        Text2 = Text2 & ","
    Next i

    '符号化したデータをファイル(textcode.txt)へ保存
    Open "textcode.txt" For Output As #1
    Print #1, Text2
    Close #1
End Sub

'*******************************************************
' ファイル(textcode.txt)のデータを読み込んで，文字を表示
'*******************************************************
Private Sub Command1_Click()
    Dim moji As String
    Dim mojicode(MAX), i As Long

    'テキストボックスを初期化
    Text1 = "": Text2 = ""

    'ファイルをオープン＆読み込み
    Open "textcode.txt" For Input As #1
    For i = 0 To MAX
        Input #1, mojicode(i)
    Next i
    Close #1

    '文字へ変換
    For i = 0 To MAX
        If Chr(mojicode(i)) <> 0 Then
            moji = moji & Chr(mojicode(i))
        End If
    Next i

    '変換した文字列を表示
    Text2 = moji
End Sub
```

《リスト2》VB用『点つぶし』の改造部分のリスト

```
'以下の定数をコメントにそって変更すると，そのデータがゲームに反映される
'(以下の設定は，写真2の場合)

Const TekiTime = 30 '敵の移動の向きが変わる頻度。小さいほどよく変わる
Const TekiHajime = 150 '敵の最初の数
Const TekiH = 1 '敵を倒せなかった時に増える敵の数
Const KAZU = 130 '自分の触手の長さ
Const RenSinPuku = 1 '連鎖が可能になる最低限の触手の振幅。大きくするほど難しくなる
Const BgmFile = "C:¥WINDOWS¥MEDIA¥Passport.mid" '音楽ファイル
Const Wt = 60 'ウェイト調整　小さいほど速くなる
```

《リスト3》iアプリ用『とりでの攻防』の改造リスト

```
※151行目以降の記述を以下のようにする

} else { // ここが151行目
    // コンピュータの発射角度を決める
    ★改造箇所(COM処理部分をコメントアウト)
    /*
    getKey();
    cr = Math.abs(rn.nextInt() % 91);
    */

    for (;;) {
        g.lock();
        // 90度を越えたら枠クリア
        if (r > 90) {
            g.setColor(g.getColorOfName(Graphics.BLUE));
            g.fillRect(ox+96, oy+6, 18, 18);
            g.setColor(g.getColorOfName(Graphics.WHITE));
            g.drawRect(ox+95, oy+5, 20, 20);
            r = 0;
        }
        g.setColor(g.getColorOfName(Graphics.AQUA));
            g.drawLine(ox+114, oy+24, ox+114-((sincos[90-r]*18)>>10),
oy+24-((sincos[r]*18)>>10));

        // コンピュータが決めた発射角度になったら発射の準備
        ★改造箇所(「if(r == cr)」を「if (getKey())」に変更)
        // if (r == cr) {
        if (getKey()) {
            x = 95 << 10;
            y = ty2 << 10;
            vx = -(sincos[90-r] >>1);
            vy = -sincos[r]-10;
            vvx = 0;
            break;
        }
        r++;
        g.unlock(true);
        wt();
    }
}

※ 以下の記述は，そのまま
```

フリーソフト

HSPde ネットワークゲーム！

Hot Soup Processer

講師:うすあじ

WEBサイト「HSPセンター」を運営するHSPファンの一人。ゼリー大好き！

第5回 レーダー作戦ゲームを作る

レーダー作戦ゲームとは

もとはコンピュータを使わないボードゲーム仲間で，９×９の海に浮かべた戦艦を狙いあうゲームです。相手の配置の様子は見えないので，それを予想しながら攻撃します（第1図）。

本講座ではHSP/O2を使ったネットワークゲームとして「レーダー作戦ゲーム」を再現します。

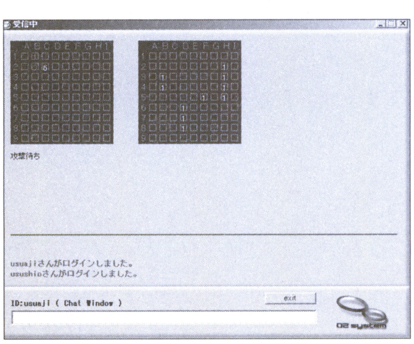
前回の「HSPdeネットワークゲームを作ろう！」では，概略やルールについて考えたので，今回はスクリプトとテクニックについて解説していきます。

ゲームの流れ

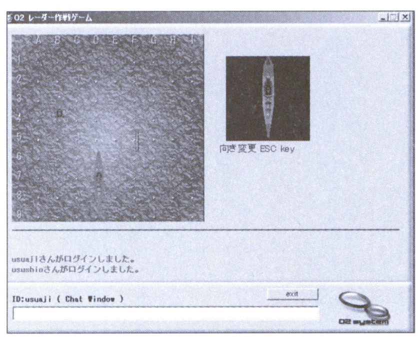
全体の流れは次のようになっています。

(1)ロビー

部屋の登録，対戦相手との待ち合わせをします。



《第1図》ゲーム画面



《第2図》船の位置を設定

(2)船の位置を設定

海をクリックして船を配置します（第2図）。

ESCキーを押すと，これから配置する船の縦横がかわります。

１マスを使う潜水艦から，４マスを使う戦艦までの４隻を配置します。４隻の配置を決定するにはスペースキーを押します。指定し直すときにはESCキーを押してください。

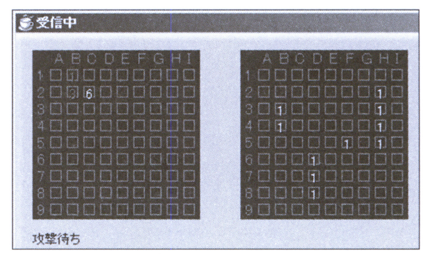
・配置マップについて

マップ表示などのために，二つの配列を用意しています。配列変数map0は船の配置の元データとして，配列変数map1は船を爆破した場合に値を変更するために使用しています。

(3)ゲームスタート

最初に部屋を設定したプレイヤー(プレイヤー0)からの攻撃を行います。

以下はプレイヤー0の動作です。



《第3図》攻撃マップと守備マップ

(4)爆弾の位置を入力

左側のマップをクリックして，攻撃する位置を決めます（第3図）。自分の海域の様子は右側に表示されます。クリック座標から計算し，0から80までの数値に変換して，対戦相手に送信します。

(5)結果表示

対戦相手から攻撃の結果が届きます。攻撃が成功すれば赤字で表示され，周辺のマスに船がいる場合には，そのマス数が表示されます。

攻撃の結果は配列に保存し，次回以降画面に表示されるようにしておきます。

(6)攻撃待ち

対戦相手からの攻撃を待ちます。

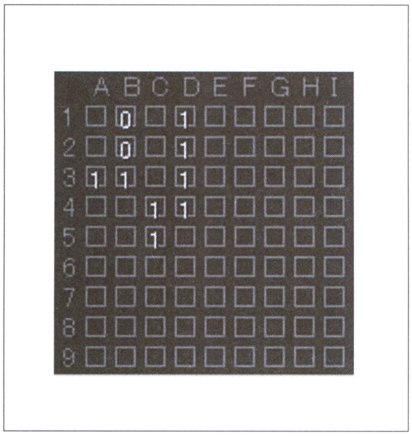
攻撃場所の情報を受け取ったら，攻撃の結果を送信します。

(7)結果表示

攻撃された結果を画面上に表示します(第4図)。

数字が1のマスは船がいるマス。

数字が0のマスはすでに爆破されたマスです。



《第4図》守備マップ

(8) ゲームオーバー

どちらかの船が全滅すればゲームオーバーです。ゲームオーバーになるとスクリプトは停止します。

再度遊ぶ場合には，チャット画面にあるEXITボタンを押して，対戦中の部屋から抜け，ゲームを終了してください。

各モードについて

《第1表》モード一覧

モード	動作
モード0	戦艦配置
モード1	攻撃データ転送
モード2	被害データ転送
モード3	状況表示
モード4	ゲームオーバー

チャットやサーバーとのやり取りの都合上，このゲームはモードという概念を使って，現在のゲームの状態を表しています。モードは変数modeに入っています（**第5図**）。

スクリプト

・座標の変換

マスの指定には，0から80までで指定する一次元的な値と，xyで指定する二次元的な値があります。一次元的な値が変数locにあり，変数xと変数yに変換したい場合には以下のようにします。

```
x=loc¥9
y=loc / 9
```

マスは9個ずつ並んでいるので，9で割った余りがマップのxになります。

¥マークは余りを得るための計算記号です。知っていると便利です。

yは9で割った答えです。

HSPでは小数点以下は無視されて，このままの式でyが得られます。

・攻撃情報の記録

画面上のマップで，今まで攻撃した位置と結果がわかる仕組みになっています（**第6図**）。

配列変数blocに攻撃位置，配列変数brcvに攻撃結果をためておき，マップ上に表示しています。

・実行時の注意

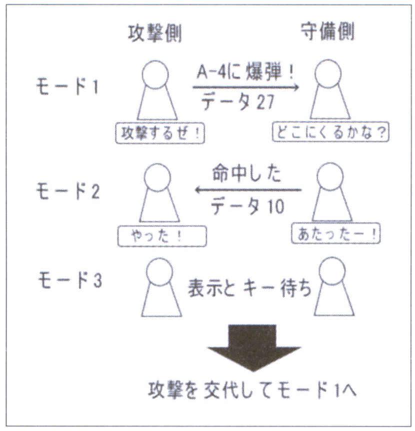
スクリプトのテストには，ローカル環境でLOBBYサーバーを動かします。

テスト用サーバー**l_server.exe**は開発キット内のbinディレクトリにあります。

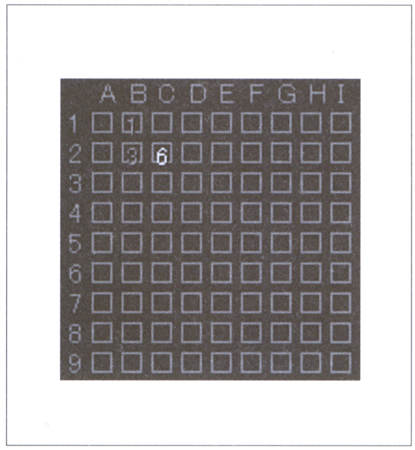
グラフィック

グラフィック（**第7,8図**）は船の配置の時に使います。ゲーム中の画面ではサイズが合わないし，情報が見づらいので，凝ったグラフィックは使いませんでした。

また，O2のボタン表示などに，標準添付のグラフィック**o2_bar.bmp**を使用します。



《第5図》フロー



《第6図》攻撃マップ



《第7図》配置時の背景グラフィック



《第8図》配置時の船舶グラフィック

まとめ

さて，駆け足で勉強してきました，**HSP/O2**を使ったネットワークゲームの世界ですが，本誌休刊にともない，今回で終わりになります。

微力ではありますが，ネットワークゲームプログラムの入り口を紹介できることができたと思います。

それでは，いつかインターネット上で皆さんの作ったネットワークゲームに出会える日を楽しみにしています。

では，お元気で。

HSP/O2について

O2はonion softwareと(有)ツェナワークスが共同で提供してるサービスです。研究とネットゲーム作成環境の普及のために，無料で一般に提供してます。

《リスト》レーダー作戦ゲームのソーススクリプト

```
;
;		レーダー作戦ゲーム
;
#include	"o2.as"
*main
		gosub *apinit
		o2init
		o2start "",-2				;ローカルサーバーで2人用
*main1
		; ロビーを開始
		o2lobby
		objsize 80,18
		pos 414,410:button "exit",*btnexit
		onexit *btnexit
*main2
		; ロビー内のループ
		o2lchat
		if o2lchat_stat : goto *main3
		o2wait 30
		goto *main2
*btnexit
		; ロビーでexitボタンを押した
		o2lexit
		end
*errtrap
		; 致命的なエラーが起こった場合
		o2error
		end
*main3
		; ロビー終了
		if o2lchat_stat<0 : goto *errtrap
*main4
		; ゲームを開始
		o2game 4	; ゲームモード開始(チャット行数を指定)
		objsize 80,18
		pos 414,410:button "exit",*gcexit
		onexit *gcexit2
		gosub *ginit
		goto *main5
*gcexit
		; ゲームを終了(ロビーに戻る)
		gosub *gterm
		o2gexit
		goto *main1
*gcexit2
		; ゲームを終了
		gosub *gterm
		o2gexit
		end
*main5
		; ゲーム内のループ
		o2gcls				; ゲームエリアをクリア
		gosub *gmain
		o2gupdate			; ゲームエリアを再描画
		o2wait 30
		o2gchat				; チャット機能
		if o2gchat_stat : goto *main6
		goto *main5
*main6
		if o2gchat_stat>0 : goto *gcexit
		goto *errtrap

;-------------------------------------------------------------------
		;		ここからゲーム固有のスクリプトになります

;-------------------------------------------------------------------
*apinit
		;		全体初期化のサブルーチン(最初の１回だけ呼ばれ
ます)
		;
		title "O2 レーダー作戦ゲーム"		; タイトル名
		return
*ginit
		;		ゲーム初期化のサブルーチン(ゲーム開始時１回だ
け呼ばれます)
;game init
		;配置ツールの初期化部分
		gosub *editstart

		dim bloc,100 ;攻撃位置
		dim brcv,100 ;攻撃結果
		bct=0	;攻撃データ数

		user0=""
		user1=""
		seNameGet user0, 0
		seNameGet user1, 1
		myid=stat:ban=0
		mode=0 ;game mode
		return
;-----------------------------------------------
*editstart
		;初期化とグラフィック読み込み
		buffer 5
		picload "umi0.bmp"
		buffer 4
		picload "all.bmp"
		;配置画面初期化
*editinit
		gsel 0
		dim sx,10
		dim sy,10
		dim sf,10
		dim map0,10,10 ;me マップ
		dim map1,10,10 ;コピー（当たると消す）
		ship=0
		flip=0
		dim shiplocx,10 ;ship loc
		dim shiplocy,10 ;ship loc
		dim shipflip,10 ;ship flip
		return
*editput
		gsel 0
		pos 16,16:gcopy 5,0,0,300,290
		repeat ship
			x=shiplocx.cnt
			y=shiplocy.cnt
			ch=cnt*2+(shipflip.cnt)+1
			pos x*30+16,y*30+16
			gosub *shipput
		loop
		color 0,0,100
		boxf 350,50,480,180
		if flip=0:pos 400,50:else:pos 350,100
		ch=ship*2+flip+1:gosub *shipput
		return
;-----------------------------------------------
*gterm
		;		終了処理のサブルーチン(最後の１回だけ呼ばれます)
		;
		return
*gmain
		;		毎フレームのサブルーチン(ゲーム中30msごとに呼ばれます)
		;
		seUserStat myid^1				; 相手がオンラインかチェック
		if stat=0 : o2gchat_stat=1 : return		; 相手がオフラインになった時
		font "MS ゴシック",15
		if mode=1:goto *game_mode1
		if mode=2:goto *game_mode2
		if mode=3:goto *game_mode3
		if mode=4:goto *game_mode4
		if mode=5:goto *game_mode5
*edittop
		gosub *editput
*cliklp
		if ship!4 {
			mx=(mousex-16)/30:my=(mousey-16)/30
			if flip=1:maxx=10-ship:maxy=10
			if flip=0:maxx=10:maxy=10-ship
			getkey key,27 ;flip処理
			pos 340,185:mes "向き変更 ESC key"
			if key {
				flip=flip^1
				gosub *editput
*clicoff
				getkey a,27
				wait 1
				if a:goto *clicoff
			}
			if (mx>0)&(mx<maxx)&(my>0)&(my<maxy){
				getkey a,1 ;マウス左
				mc=0
				cx=mx-1:cy=my-1
				repeat ship+1
					c=map0.cx.cy
					if c:mc=1
					if flip=0:cy++:else:cx++
				loop
				if (a=1)&(mc=0) {
					cx=mx-1:cy=my-1
					repeat ship+1
						map0.cx.cy=1 ;map write
						shiplocx.ship=mx
						shiplocy.ship=my
						shipflip.ship=flip
						if flip=0:cy++:else:cx++
					loop
					sx.ship=mx-1:sy.ship=my-1
					sf.ship=flip:ship++
				}
			}
		} else {
			pos 340,185
			mes "OK space key"
			mes "再配置 ESC key"
			getkey a,27 ;ESC
			if a:gosub *editinit
			getkey a,32 ;space
			if a {
				repeat 9
					yct=cnt
					repeat 9
						a=map0.cnt.yct
						map1.cnt.yct=a
					loop
				loop
				gsel 0:o2gcls:mode=1
			}
		}
		return
```

```
*game_mode1
		; mode1
		if myid=ban {
		;攻撃側入力
			gsel 0
			gosub *aiteput
			font "MS ゴシック",15
			myte=-1
			mx=mousex-16/16:my=mousey-16/16
			title "クリックで攻撃 "+mx+"/"+my
			getkey key,1 ;マウス左
			if (mx>0)&(mx<10)&(my>0)&(my<10)&(key=1) {
				myte=(my-1)*9+mx-1
				h=mx:v=my:aa="*"
				gosub *puttxts
;				redraw 1
				o2wait 10
			}
			if myte!-1 :a=myid^1 : seDataPut myte,a:mode=2
		} else {
		;守備側データ受信
			title "受信中"
			gosub *aiteput
			color 0,0,0:pos 16,190
			mes "攻撃待ち"
			seDataStat
			if stat > 0 {
				seDataGet data
				mode=2:wait_ct=50
			}
		}
		return
*game_mode2
		; mode2
		if myid=ban {
		;攻撃側結果受信
			title "受信中"
			gosub *aiteput
			font "MS ゴシック",15
			color 0,0,0:pos 16,190
			mes "結果の返信待ち"
			seDataStat
			if stat > 0 {
				seDataGet data
				bloc.bct=myte ;結果のストック
				brcv.bct=data:bct++ ;結果のストック
				mode=3:wait_ct=50
			}
		} else {
		;守備側データ送信
			;hit check(data)
			color 0,0,0:pos 16,190
			mes "データ送信中 "
			gosub *aiteput
			bx=data ¥ 9
			by=data / 9
			if map0.bx.by=1:kekka=10:else:kekka=0
			map1.bx.by=0		;game overチェック用のmap1
			i=0
			repeat 80
				loc=cnt
				x=loc ¥ 9:y=loc / 9
				i+=map1.x.y;game overチェック用のmap1
			loop
			if i=0:kekka=90 ;game over
			x=bx-1:y=by-1:gosub *oshii
			x=bx  :y=by-1:gosub *oshii
			x=bx+1:y=by-1:gosub *oshii
			x=bx-1:y=by  :gosub *oshii
			x=bx+1:y=by  :gosub *oshii
			x=bx-1:y=by+1:gosub *oshii
			x=bx  :y=by+1:gosub *oshii
			x=bx+1:y=by+1:gosub *oshii
			a=myid^1 : seDataPut kekka,a
			if i=0:mode=4:else:mode=3:wait_ct=50
		}
		return
*oshii
		if x<0:return
		if x>9:return
		if y<0:return
		if y>9:return
		kekka+=map0.x.y
		return
*game_mode3
		; mode3
		if myid=ban {
		;攻撃側データ表示
			gosub *aiteput
			font "MS ゴシック",15
			color 0,0,0:pos 16,190
			mes "受信データ "+data
			hit=data/10 ;hit 0 or 1 or 9
			if hit>0:mes "ヒットした！"
			wait_ct--
			stick key,15,1
			if (wait_ct=0)|(key) {
				mode=1:ban=ban^1
			}
			if hit=9:mes "ゲームオーバー！":wait_ct=50:mode=4
		} else {
		;守備側表示
			gosub *aiteput
			color 0,0,0:pos 16,190
			hit=kekka/10
			if hit=1:mes "Hit 損害を受けた"
			if hit=0:mes "Safe 被害なし"
			if hit=9:mes "Game over ゲームオーバー" :mode=4:return
			wait_ct--
			stick key,15,1
			if (wait_ct=0)|(key) {
				mode=1:ban=ban^1
			}
		}
		return
*game_mode4
		; mode4
		if myid=ban {
;勝った！
			gosub *aiteput
			color 0,0,0:pos 16,190
			mes "全艦爆撃。勝ちました。"
		} else {
;負けた！
			gosub *aiteput
			color 0,0,0:pos 16,190
			mes "負けました。"
		}
		return
*game_mode5
		return
;サブルーチン
;今までのデータを表示
*aiteput
		gsel 0
		hof=16:vof=16:gosub *stage
		repeat bct
			loc=bloc.cnt
			h=(loc ¥ 9)+1:v=(loc / 9)+1
			aa=brcv.cnt:gosub *puttxts
		loop
		hof=216:vof=16:gosub *stage
		repeat 81
			loc=cnt
			x=loc ¥ 9:y=loc / 9
			h=(loc ¥ 9)+1:v=(loc / 9)+1
			if map0.x.y {
				aa=map1.x.y
				gosub *puttxts2
			}
		loop
		return
*stage
		font "MS ゴシック",15
		color 0,0,100
		boxf hof,vof,hof+160,vof+160
		color 128,128,128
		;数字部分
		repeat 9,1
		 h=0:v=cnt:a=" "+cnt:pos h*16+hof,v*16+vof:mes a
		loop
		;文字部分
		b="A"
		repeat 9
		 char='A'+cnt:poke b,0,char
		 v=0:h=cnt+1:pos h*16+hof,v*16+vof:mes " "+b
		loop
		;ドット部分
		repeat 9
		 vct=cnt+1
		 repeat 9
		 v=vct:h=cnt+1:pos h*16+hof,v*16+vof
		 mes "□"
		 loop
		loop
		return
*shipput
		csc=4:gmode 2
		if ch=1:gcopy csc,0,0,32,32		;潜水艦縦 1
		if ch=2:gcopy csc,224,0,32,32		;潜水艦横 2
		if ch=3:gcopy csc,32,0,32,64		;駆逐艦縦 3
		if ch=4:gcopy csc,192,32,64,32		;駆逐艦横 4
		if ch=5:gcopy csc,64,0,32,96		;重巡洋艦縦 5
		if ch=6:gcopy csc,160,64,96,32		;重巡洋艦横 6
		if ch=7:gcopy csc,96,0,32,128		;戦艦縦 7
		if ch=8:gcopy csc,128,96,128,32	;戦艦横 8
		return
*puttxts
		font "MS ゴシック",12
		tmpx=h*16+20:tmpy=v*16+20
		bb=aa:int bb
		color 0,0,0:pos tmpx+1,tmpy+1
		if bb < 9 :mes aa:else:color 0,0,0:mes bb¥10
		color 255,255,255:pos tmpx,tmpy
		if bb < 9 :mes aa:else:color 255,0,0:mes bb¥10
		return
*puttxts2
		font "MS ゴシック",12
		tmpx=h*16+220:tmpy=v*16+20
		color 0,0,0:pos tmpx+1,tmpy+1:mes aa
		color 255,255,255:pos tmpx,tmpy:mes aa:return
```

.NETでゲームを作ろう!

.NET GAME PROGRAMMING ドットネットゲームプログラミング

最終回 タイピングゲーム

7

赤坂　玲音

.NET 開発環境

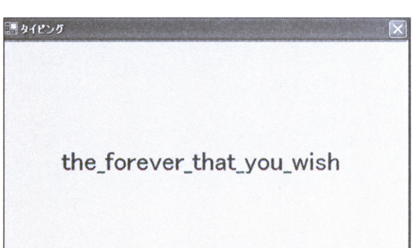
ここでは.NET アプリケーションの開発方法を，C#言語とVisual Basic.NETを使って，ゲームを作りながら解説していきます。 .NETアプリケーションの開発には，Microsoft が提供する .NET 開発環境を使ってプログラムします。 通常，.NET アプリケーションの開発にはMicrosoft Visual Studio.NET (VS.NET)を使います。 .NET開発環境の詳細については3月号を参照ください。

ゲーム仕様

今回は，きわめて単純なタイピングゲームの基盤を作成しながら，ゲーム開発の基本的ノウハウと .NET の入出力について学習します。

ゲームは，**写真1**のように画面中央に入力するべき英数字が表示され，キーボードから文字を入力すると，左から順に文字が消えていくというものです。

表示された文字をすべて入力すると，乱数によって次の文字列が選択され，画面に表示されます。得点やミスタイプの処理は作成しないので，読者が自由に拡張できる形でコーディングしたいと思います。



《写真1》タイピングゲームの画面

入出力とスクリプト

ベーマガの投稿プログラムのような小規模のゲームは例外ですが，商業レベルのゲームプログラミングでは，定数をコード内に書いてはならないという鉄の掟があります。

つまり，プログラムコードの中に x = 10 というような整定数を使った式や，WriteLine("えぅ～，あんまんは命の源だよぉ") といった文字定数を書かないのです。もちろん，コード内における定数の役割や意味によっても，書くか書かないかという議論に発展するかもしれませんが，原則としてハードコードは避けるべきです。特に，ゲームの流れを制御するために使われる定数をコード内に書くようなことがあってはなりません。

では，数値や文字情報はどのように扱うべきなのでしょうか。通常，円周率のような固定的な情報は，マクロや const キーワードを使って変更不可能な変数として扱います。 実際に.NET の標準クラスライブラリである System.Math クラスは定数 π を const フィールド PI として公開しています。式に 3.14… と定数を直接記述するよりも，Math.PIを使ったほうが整合性，柔軟性において，はるかに理にかなっているでしょう。円周率は変化することがないと思われますが，ゲーム開発段階の場合，設定の変更やスクリプタの突然のアイデアで頻繁に更新する必要がある定数も存在します。

例えば，キャラクターの台詞やオブジェクトの位置情報，色，フォント，画像ファイルなどは，開発中に自由に変更可能でなければなりません。こうした情報をハードコードしてしまうと，変更のたびにコンパイルする必要があります。

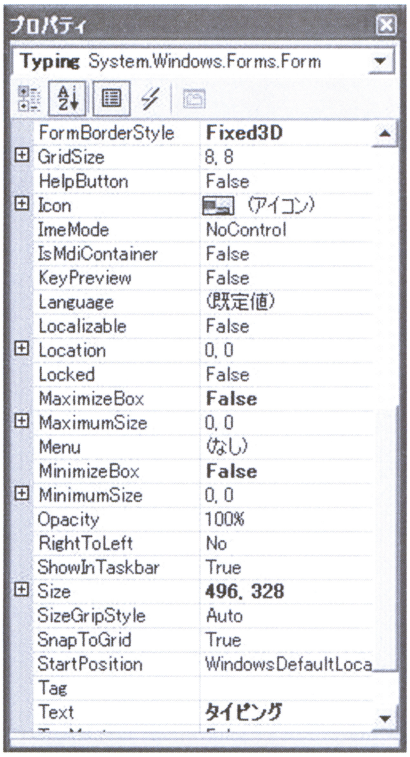
個人レベルのプログラムではあまり感じないかもしれませんが，数万行に及ぶ中規模以上のプロジェクトでは，完成したプログラムを再コンパイルするのは危険な行為です。もちろん，デバッグもやり直さなければならないでしょう。

こうした問題を避ける一般的な方法が，リソールやスクリプトなどの形で，プログラムの外部に定数を追いやることです。プログラムは実行時にデータを読み込み，ゲームを進行するのです。そうすれば，再コンパイルする必要なく，スクリプトを書き換えるだけでゲームを自由に制御することができます。

タイピングゲームで考えられる定数は，入力対象文字列の配列や，入力待ち時間，文字に対する得点などでしょう。

フォームの作成

まず，ゲームの土台となるフォームを

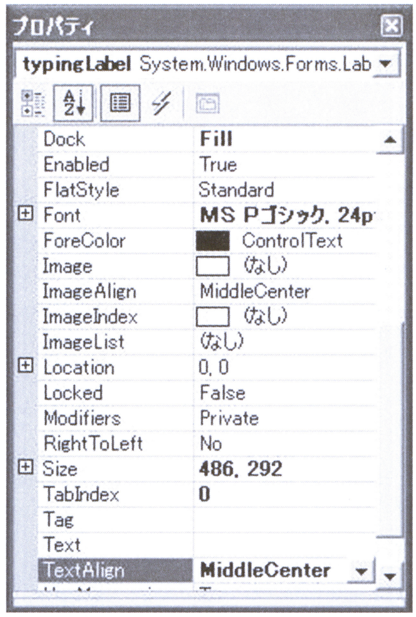


《写真2》Form のプロパティ

作りましょう。といっても，今回のゲームはフォームにラベル コントロールを貼り付けただけの，いたって簡単なものです。VS.NET 環境であればデザイナを使って，SDK であれば直接 System.Windows.Forms.Form クラスを継承して**写真2**のようにプロパティを設定してください。

次に，このフォームに文字列を描画するラベルを貼り付けます。SDK であれば，System.Windows.Forms.Labelクラスをインスタンス化して，フォームに追加してください。 ラベルは，常にフォームのクライアントサイズと同じサイズになるように Control.Dockプロパティを DockStyle.Fill に設定します。テキストの表示位置は好みでかまいませんが，中央が適切だと思われるので Label.TextAlignプロパティ(**写真3**)を ContentAlignment.MiddleCenter に設定してください。ラベルのフォントは，この場では好みのものを適当に指定することにしましょう。

ゲームの性質上，フォントや文字色は重要な情報となるため，できることなら外部ファイルとして読み込んだり，readonlyの非公開フィールドとして扱うべきであると考えることもできます。余裕のある方は，そのように改良しても良いでしょう。



《写真3》Labelのプロパティ

文字列の表示

次に，入力対象文字列をラベルに表示させるメソッドを作りましょう。このメソッドは，乱数を使って配列から適当な文字列を選択します。配列は文字列配列でもかまいませんが，この場は System.Collections.ArrayList クラスを使った動的な配列を採用します。

ArrayList を使えば，あらゆるオブジェクトを入力対象文字列として扱うことができますし，追加や削除を自由に行えます。あとは，Label.Text プロパティに対象文字列を設定するだけですが，ちょっと細工します。

英文の空白を明示的に表現するために，スペースをアンダーラインに置換えましょう。 文字列内の特定の文字を置換えるにはString.Replace() メソッドを使います。

```
[Serializable] public string Replace(char oldChar , char newChar);
```

oldCharには置換するUnicode 文字，newCharには出現するすべてのoldCharを置換えするUnicode文字を指定します。このメソッドは，インスタンスと等価な置換え後の String を返します。

入力処理

次に，キーボードの入力を処理してタイピング ゲームを実現しましょう。 この作業もなんら難しいものではありません。

現在，ラベルに表示されている英数文字の先頭を調べ，それと等しいキーの入力であれば，ラベルの先頭文字を消せばよいのです。ただし，ラベルの文字がすべて消えた場合は，再びNextLine() メソッドを呼び出します。

キーボードからの入力は，Control.OnKeyPress() メソッドをオーバーライドすることで取得することができます。

```
protected virtual void OnKeyPress
(KeyPressEventArgs e);
```

System.Windows.Forms.KeyPressEventArgs クラスは，入力したキーによっ

《コード1》NextLine() メソッド C#

```
private void NextLine()
{
        Random rand = new Random();
        string temp = list[rand.Next(list.Count)].ToString();

        this.typingLabel.Text = temp.Replace(' ' , '_');
}
```

《コード2》NextLine() メソッド VB .NET

```
Private list As ArrayList = New ArrayList()
Private Sub NextLine()
    Dim rand As Random = New Random()
    Dim temp As String = list(rand.Next(list.Count)).ToString()

    Me.typingLabel.Text = temp.Replace(" "c, "_"c)
End Sub
```

て作成された文字を格納しています。この情報を調べれば，ラベルの先頭の文字と入力されたキーが等しいかどうかを調べることができます。

System.Object
¦
System.EventArgs
¦
System.Windows.Forms.KeyPressEventArgs

[ComVisible(true)] public class KeyPressEventArgs : EventArgs

文字はKeyPressEventArgs.KeyCharプロパティから取得することができます。

[ComVisible(true)] public char KeyChar {get;}

このプロパティとラベルの先頭の文字を比較するには，Label.Text から先頭の文字を抽出する必要があります。そのためには文字列を配列化すればよいのでString.ToCharArray() メソッドを使います。

[Serializable]public char[] ToCharArray();

この char 型配列の先頭を調べれば，タイピング処理を実現することができるでしょう。ただし，スペースがアンダーラインに置き換えられていることを忘れてはなりません。アンダーラインの入力は無視し，スペースが入力された場合はアンダーラインと等しくなるように比較してください。

入力された文字とラベルの先頭が等しければ，ラベルの先頭の文字を削除する必要があります。文字列を部分的に抽出するには String.Substring() メソッドを利用します。ただし，文字列の長さが1の場合は空文字になってしまうため，NextLine() メソッドを呼び出して新しい文字列をセットするように仕組む必要があります。

[Serializable]public string Substring

《コード3》入力処理 C#

```
protected override void OnKeyPress(KeyPressEventArgs e)
{
        char temp = this.typingLabel.Text.ToCharArray()[0];

        if ((temp == '_' && e.KeyChar == ' ') || (e.KeyChar == temp && temp !=
'_'))
        {
                if (this.typingLabel.Text.Length == 1) NextLine();
                else this.typingLabel.Text =
this.typingLabel.Text.Substring(1);
        }
        base.OnKeyPress(e);
}
```

《コード4》入力処理 VB .NET

```
Protected Overrides Sub OnKeyPress(ByVal e As
System.Windows.Forms.KeyPressEventArgs)
    Dim temp As Char = Me.typingLabel.Text.ToCharArray()(0)

    If (temp = "_"c And e.KeyChar = " "c) Or (e.KeyChar = temp And temp <> "_"c)
Then
        If Me.typingLabel.Text.Length = 1 Then
            Me.NextLine()
        Else : Me.typingLabel.Text = Me.typingLabel.Text.Substring(1)
        End If
    End If

    MyBase.OnKeyPress(e)
End Sub
```

(int startIndex);

startIndex には文字列の開始文字位置を指定します。このメソッドは，startIndex から文字列の終端までを部分文字として返します。

Substring() メソッドに 1 を渡せば，文字列の先頭文字を排除することができるでしょう。

リストの初期化

ゲームの基本的な制御を行うコードの準備は整いました。あとは ArrayListのインスタンスを正しく初期化して，入力対象文字列を用意するだけです。

ArrayList の初期化はコンストラクタなど，プログラムコードのあらゆる場所で行うことができますが，ここで再び「定数をハードコードしない」という掟を思い出しましょう。

プログラムコードに直接，入力対象文字列を記述するのはナンセンスです。となれば，ゲームの問題となる入力対象文字列の配列は外部ファイルとして用意することになります。

入力対象文字列をテキストファイルとして実行ファイルの外に配置すれば，テキストエディタでファイルを編集するだけで，自由にゲームを拡張できることになります。

テキストファイルの読み込みは，さまざまなアプローチが考えられますが，最も簡単なのはSystem.IO.StreamReaderクラスを利用した方法でしょう。

System.Object
¦
System.MarshalByRefObject
¦
System.IO.TextReader
¦
System.IO.StreamReader

[Serializable] public class StreamReader : TextReader

通常，テキストファイルから文字データを読み込む場合はこのクラスを使います。StreamReader クラスのコンストラクタはオーバーロードされていますが，今回はファイルのパスを引数に渡す一番簡単な方法でインスタンスを生成しましょう。

《コード5》初期化処理

```
private const string DATAFILE = "Typing.txt";
public Typing()
{
        InitializeComponent();

        StreamReader reader = new StreamReader(DATAFILE);
        for(string temp ; (temp = reader.ReadLine()) != null ;) list.Add(temp);

        NextLine();
}
```

《コード6》初期化処理 VB .NET

```
Private Const DATAFILE As String = "Typing.txt"

Public Sub New()
    MyBase.New()

    InitializeComponent()

    Dim reader As StreamReader = New StreamReader(DATAFILE)
    While True
        Dim temp As String = reader.ReadLine()
        If temp = Nothing Then
            Exit While
        Else : list.Add(temp)
        End If
    End While

    NextLine()
End Sub
```

《データ1》Typing.txt ファイルの例

```
forget me not
snow
muv luv
kanon
to heart
```

[Serializable] public StreamReader (string path);

path にはテキストファイルへのパスを指定します。相対パスでもかまいません。StreamReader のインスタンスを無事に生成することができれば，あとは StreamReader.ReadLine() メソッドを使って文字列を読み込みましょう。ReadLine() メソッドは，最初の行から順に文字列を読み込んでくれます。

[Serializable] public override string ReadLine();

ReadLine() メソッドは，いったいどの行を読み込んでくれるのでしょうか？この疑問を解決するには，ストリームについて少し説明する必要があります。すべての入出力はストリームという概念によって，統一された操作を実現しています。

私たちプログラマーは，ストリームに入出力対象データを関連づけて，その後にストリームを操作するという方法でデータとプログラムを結びつけることができるのです。ストリームの恩恵は偉大なものです。

開発者はストリームとデータを関連づけることによって，入出力対象となっているデータがどこに存在するのかを，まったく気にする必要がなくなるのです。ストリームは主記憶装置のアドレスを指しているかもしれませんし，ネットワーク上の別のコンピュータのリソースに関連づけられているかもしれません。

相手がプリンタでも，ディスプレイでも，そしてファイルでも，ストリームに関連づけることによって，統一された方法でデータの入出力を行うことができるのです。

StreamReader のインスタンスは，デフォルトでストリームの先頭を指しています。その後，読み込み処理を行うたびにストリームの位置を自動的に進めてくれるのです。つまり，ReadLine() メソッドを呼び出した場合，ストリームは次の行に自動的に移動するのです。ReadLine() メソッドは，現在のストリームの位置から行の末尾までを読み込み，ストリームの位置を次の行に進めます。しかし，出力ストリームとは異なり，入力ストリームには明確な末尾が存在します。ファイルの場合，ファイルの終端にたどり着いたことを調べなければなりません。

幸い，ReadLine() メソッドは入力ストリームの末尾に到達するとnullを返すので，今回はこれを利用してテキストファイルから文字列を読み込みます。あとは文字列をArrayListに追加するだけです。

コード5とコード6は，このプログラムのコンストラクタです。Typing.txtという名前のテキストファイルから各行を読み込んでリストに追加していることを確認してください。InitializeComponent()は，フォームの初期化を行うためのメソッドです。テキストファイルには，タイピングの問題となる文字列を一行ずつ指定します。あとは，このファイルを実行ファイルと同じフォルダに Typing.txt という名前で保存してください。

ゲームの完成

これで，タイピングゲームの基盤は完成です。この状態では，得点の管理や時間制限などがまったくないので，ゲームとしては成り立ちませんが，コアであるタイピング処理が完成したので，残るは肉づけだけです。

以前解説したタイマーを使えば，時間制限なども簡単に実現できるでしょう。何度もしつこいのですが，できるかぎり定数をコードに直接書かない癖をつけることです。

例えば，このタイピングゲームを発展させて，物語を付加してアドベンチャーパートをプログラムする場合，やはり，

スクリプトとして外部にデータを配置する必要があります。アドベンチャーやノベル系のゲームは，処理をスクリプト化する代表的なプログラミングの例でしょう。ストーリーやキャラクターの位置を変更するたびにコンパイルするのはあまりにも危険です。

しかし，ここでもうひとつ大きな問題が浮上します。

今回のタイピングゲームも同様ですが，テキストデータとしてプログラムの外に情報を配置した場合，ユーザーが勝手に編集してしまう危険性があります。編集が容易なのはスクリプタにとって重要ですが，データが簡単に覗かれてしまうようでは，パソコンゲームの場合，売り物になりません。

スタンダードな開発手法では，開発段階ではテキストファイルとして読み込み，最終的にはバイナリファイルとして読み込む方法が採用されています。こうした開発手法を導入する場合，スクリプトの前にバイナリで表現された中間言語仕様を決める必要があります。

そして，その中間言語の上にテキストで表現されたスクリプトの仕様をまとめ，スクリプトをバイナリの中間言語に翻訳するコンパイラも用意しなければなりません。

開発規模が大きくなれば，スクリプトでも管理が難しくなるため，さらにスクリプトを自動生成する開発ツールも作成する場合もあります。柔軟で汎用的なゲームを開発するには，このように多くのツールの開発が必須となることでしょう。

少し話が大きくなってしまいましたが，どんなに大規模なプロジェクトも，小さく単純なプログラム部品の集合でできていることを忘れないで下さい。明確な設計理念をもって開発すれば，個人でも企業プロジェクトに負けない質の高いソフトウェアを開発できるはずです。 そして，Microsoft .NETは，その開発の強力なパートナーとなってくれることでしょう。

《コード7》タイピングゲーム C#

```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.IO;

namespace Typing
{
    public class Typing : System.Windows.Forms.Form
    {
        private System.Windows.Forms.Label typingLabel;

        private ArrayList list = new ArrayList();
        private const string DATAFILE = "Typing.txt";    //データファイルのパス

        private System.ComponentModel.Container components = null;

        public Typing()
        {
            InitializeComponent();

            StreamReader reader = new StreamReader(DATAFILE);
            for(string temp ; (temp = reader.ReadLine()) != null ;)
list.Add(temp);

            NextLine();
        }

        private void NextLine()
        {
            Random rand = new Random();
            string temp = list[rand.Next(list.Count)].ToString();

            //スペースはアンダーラインに置き換える
            this.typingLabel.Text = temp.Replace(' ', '_');
        }

        protected override void Dispose( bool disposing )
        {
            if( disposing )
            {
                if (components != null)
                {
                    components.Dispose();
                }
            }
            base.Dispose( disposing );
        }
```

```
		}

		#region Windows Form Designer generated code

		private void InitializeComponent()
		{
			this.typingLabel = new System.Windows.Forms.Label();
			this.SuspendLayout();
			// typingLabel
			this.typingLabel.Dock = System.Windows.Forms.DockStyle.Fill;
			this.typingLabel.Font = new System.Drawing.Font(
				"ＭＳ Ｐゴシック", 24F,
System.Drawing.FontStyle.Regular,
				System.Drawing.GraphicsUnit.Point,
((System.Byte)(128))
			);
			this.typingLabel.Name = "typingLabel";
			this.typingLabel.Size = new System.Drawing.Size(486, 292);
			this.typingLabel.TabIndex = 0;
			this.typingLabel.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter;
			// Typing
			this.AutoScaleBaseSize = new System.Drawing.Size(5, 12);
			this.ClientSize = new System.Drawing.Size(486, 292);
			this.Controls.AddRange(new System.Windows.Forms.Control[]
{this.typingLabel});
			this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
			this.MaximizeBox = false;
			this.MinimizeBox = false;
			this.Name = "Typing";
			this.Text = "タイピング for C#";
			this.ResumeLayout(false);

		}
		#endregion

		[STAThread] static void Main()
		{
			Application.Run(new Typing());
		}

		protected override void OnKeyPress(KeyPressEventArgs e)
		{
			char temp = this.typingLabel.Text.ToCharArray()[0];

			//アンダーラインはスペースが入力されると消える
			if ((temp == '_' && e.KeyChar == ' ') || (e.KeyChar == temp &&
temp != '_'))
			{
				if (this.typingLabel.Text.Length == 1) NextLine();
				else this.typingLabel.Text =
this.typingLabel.Text.Substring(1);
			}
			else
			{
				//タイプミスが発生した時のコードをここに記述します。
			}

			base.OnKeyPress(e);
		}
	}
}
```

《コード8》タイピングゲーム VB.NET

```
Imports System
Imports System.Drawing
Imports System.Collections
Imports System.ComponentModel
Imports System.Windows.Forms
Imports System.Data
Imports System.IO

Public Class Typing
    Inherits System.Windows.Forms.Form

#Region " Windows フォーム デザイナで生成されたコード "

    Private list As ArrayList = New ArrayList()
```

```
Private Const DATAFILE As String = "Typing.txt"

Public Sub New()
    MyBase.New()

    ' この呼び出しは Windows フォーム デザイナで必要です。
    InitializeComponent()

    Dim reader As StreamReader = New StreamReader(DATAFILE)
    While True
        Dim temp As String = reader.ReadLine()
        If temp = Nothing Then
            Exit While
        Else : list.Add(temp)
        End If
    End While

    NextLine()
End Sub

Private Sub NextLine()
    Dim rand As Random = New Random()
    Dim temp As String = list(rand.Next(list.Count)).ToString()

    'スペースはアンダーラインに置き換える
    Me.typingLabel.Text = temp.Replace(" "c, "_"c)
End Sub

Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
    If disposing Then
        If Not (components Is Nothing) Then
            components.Dispose()
        End If
    End If
    MyBase.Dispose(disposing)
End Sub

Private components As System.ComponentModel.IContainer

Friend WithEvents typingLabel As System.Windows.Forms.Label
 Private Sub InitializeComponent()
    Me.typingLabel = New System.Windows.Forms.Label()
    Me.SuspendLayout()
    '
```

```
        'typingLabel
        '
        Me.typingLabel.Dock = System.Windows.Forms.DockStyle.Fill
        Me.typingLabel.Font = New System.Drawing.Font("ＭＳ Ｐゴシック", 24.0!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(128, Byte))
        Me.typingLabel.Name = "typingLabel"
        Me.typingLabel.Size = New System.Drawing.Size(486, 292)
        Me.typingLabel.TabIndex = 0
        Me.typingLabel.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'Typing
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 12)
        Me.ClientSize = New System.Drawing.Size(486, 292)
        Me.Controls.AddRange(New System.Windows.Forms.Control() {Me.typingLabel})
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.Fixed3D
        Me.MaximizeBox = False
        Me.MinimizeBox = False
        Me.Name = "Typing"
        Me.Text = "タイピング for VB.NET"
        Me.ResumeLayout(False)

    End Sub

#End Region

    Protected Overrides Sub OnKeyPress(ByVal e As System.Windows.Forms.KeyPressEventArgs)
        Dim temp As Char = Me.typingLabel.Text.ToCharArray()(0)

        'アンダーラインはスペースが入力されると消える
        If (temp = "_"c And e.KeyChar = " "c) Or (e.KeyChar = temp And temp <> "_"c) Then

            If Me.typingLabel.Text.Length = 1 Then
                Me.NextLine()
            Else : Me.typingLabel.Text = Me.typingLabel.Text.Substring(1)
            End If
        Else
            'タイプミスが発生した時のコードをここに記述します。
        End If

        MyBase.OnKeyPress(e)
    End Sub
End Class
```

パソコン認定試験　対策講座

連載　第７回
パソコンリテラシ＆ネットワーク

パソコン認定試験主催：社団法人パーソナルコンピュータユーザ利用技術協会

はじめに

社団法人パーソナルコンピュータユーザ利用技術協会は，パソコンユーザの利用技術に関する調査，研究，普及啓発，情報の収集・提供および認定試験の実施，そして内外機関との交流，協力を行うことにより，情報リテラシの涵養を図り，わが国情報化社会の健全な発展に寄与することを目的に，平成3年5月に経済産業省認可の公益法人として設立された，全国を代表する唯一の機関です。

その一環として，パソコン利用技術向上のために，パーソナルコンピュータ利用技術認定試験（略称：パソコン認定試験）を全国規模で実施するなど，その教育的内容は高く評価されています。特に，パソコン認定試験合格者は，文部科学省が実施する大学入学資格検定の受検科目（情報関係基礎）が免除されます。

また，平成15年度から高等学校普通教科「情報」が新設されるにあたって，どのような知識・利用技術を身につけたらよいのか，利用者自身が主体的に情報を活用していく能力，すなわち，情報リテラシ（情報活用能力）の涵養を図るとともに，小中高校生の育成と情報化社会の健全な発展に寄与することを目指しています。

パソコン認定試験にチャレンジ！

パソコン認定試験の4級受験者のうち，高校生は，14,948名が受験し，合格者8,574名（合格率57.7%），3級は，4,663名が受験し，1,681名（合格率36.0%）が合格しています。

そのパソコン認定試験は，次回（第40回），平成15年7月13日(日)に4級，3級を実施します。

今回は，パソコン認定試験4級の受験対策として，パソコンやネットワークを利用する際，知っておくべき知識に関する過去問題を掲載し，解説しました。

あなたのチカラ試しに，次の問題を解いてみましょう！そして，パソコン認定試験本番に挑戦してみましょう！！

例題１　コンピュータで応用プログラムを使用する際に生ずる問題点に関する次の記述中の(１)～(10)に入れるべき最も適切な字句を解答群の中から選べ。

a）最近の応用プログラムでは1,000ページを超える親切なマニュアルが付属しているものもあるがマニュアルらしいものが付属していないものも増加している。大量に販売している製品の中には，プログラム中のヘルプ機能を充実させ，入門書的なものは市販の入門書又は解説書に任せようというものが増えている。ユーザが使用上で困ったとき，知りたいことを表す [(1)] が分かっているとヘルプ機能を使って調べやすいが，知りたいことを短い言葉でどう表現したらよいか分からないときは，途方にくれる。このようなとき，ヘルプの検索機能の中に，言葉の検索が上位概念から [(2)] へと構造的にたどれるように構成されていると分かりやすい。

b）文章作成プログラムを使って文章を作り，同じアプリケーションプログラムで印刷する際，参照文献の番号を上付きの文字にして付けたいことがあり，更に数式を書きたいことがある。多くの場合，数式に複雑な記号を使わないことが多い。例えば，$ax^3+bx^2+cx+d=0$と入力したい場合がある。数式編集機能を内蔵している文章作成プログラムも多いが，作業用記憶域の設定に問題があるなどの理由で数行の数式すら入力できないものもある。この場合，フォントの設定を変える， [(3)] の機能を使うなどすれば解決できることがある。

c）購入した応用プログラムを使って見ると，思いもかけないエラーメッセージに驚かされることがある。“不正な処理をしたので計算機の動作を終了します。”，“作業用の記憶領域が足りないので他の実行中のプログラムを終了してから書込みの作業をしてください。”などである。別に不正な処理をした覚えも無いし，他に実行しているプログラムも無いと，どうすれば今使っている作業領域を開放できるか分からず困ってしまう。

どんなプログラムにも必ず [(4)] がありうるものである。応用プログラムが利用者の行うすべての誤りを監視してく

れていると，誤った使い方をすると見付けて適切なメッセージを表示することができる。しかし，バグなどの原因で検出されないと，[(5)]によって不具合として検出されることになる。オペレーティングシステムでは，誤りの原因が個々の応用プログラムレベルでなく，実行される機械語命令レベルの不具合と見なされるので，エラーメッセージは，[(6)]にとって分かりにくくなる。しかもどうすれば，その現象を避けて使えるかが分からないことが多い。何時間もかけて作成した文書が，ファイルに格納できなくなり利用者は途方にくれることになる。これを防ぐには購入したらすぐ使いこなし，不具合が見つかったら早期に[(7)]に聞くのがよい。購入後時間が経つと無料で質問に答えてくれないところがある。

d）コンピュータは，暗箱となって中身が見えないようになってきている。購入した応用プログラムが，その仕様書のとおりに機能しないとき，[(8)]に誤りがあるのか，オペレーティングシステムに誤りがあるのか，ハードウェアに欠陥があるのかがプログラムを使う側には分かりにくい。ハードウェア販売業者は，販売したコンピュータ及び[(9)]に装着したオペレーティングシステムの不具合だけに責任をもつところが多い。オペレーティングシステムを新しい版に更新すると，原則として保証の対象外となる。これはコンピュータが極めて安価になり，家庭電化製品並みになったこと及びオペレーティングシステムは他社で開発しているなどの理由による。

e）オペレーティングシステムが効率よく動くために，更に，応用ソフトウェアが動くためには，一定量の主記憶容量が求められることがある。また，応用プログラムを導入するには，[(10)]に応用プログラムで必要とする容量の空いている領域が必要なことがある。最近のパーソナルコンピュータには100Gバイト近い容量のハードディスクが実装されている場合が多いので容量が足りないことは少ない。

解答群

01　応用プログラム	02　ハードディスク	03　CD-ROM
04　下位概念	05　中位概念	06　上付き，下付き
07　オペレーティングシステム	08　コンパイラ	09　販売時
10　容量が足りない	11　欠陥	12　情報を消去
13　信頼性	14　キーワード	15　利用者
16　発売元		

[第38回(2002年7月実施)パソコン認定試験４級より]

例題１の解説

アプリケーションソフトウェアを使っていると遭遇する問題点と，その対処方法についての出題です。まだ経験していないことが書かれている場合でも，今後の参考になるはずです。

a）以前のソフトウェアには膨大なマニュアルが付属していましたが，最近では，OSもアプリケーションソフトウェアも簡単な手引き程度のものしか付属しないようになってきました。パッケージが軽量，小型化し，マニュアル印刷費用がかからない分低価格になったことは，ユーザにとっては歓迎すべきことといえます。そして，操作方法などはオンラインヘルプにより画面から調べられるようになりました。しかし，ヘルプの機能が充実していないと，知りたいことがなかなか調べられないことも起こります。知りたいことをキーワードで検索する機能だけでは，キーワードが一致しない限り，目的の情報に行き着かないことになります。そこで，質問形式で入力するとそれに関するトピックが一覧表示されるようにになっていたり，階層型の目次形式で上位から下位に用語の概念が細分化されて並んでいたりすると使いやすいものになります。一方，ソフトウェアの使い方やノウハウは，市販の書籍を購入する形態に変わってきました。OSのオフィシャルマニュアルすら購入しなけ

れば手に入らなくなりました。[(1)] には“キーワード”が，[(2)] には“下位概念”が入ります。

b）文書作成プログラム（ワープロソフトウェア）ではたいていの場合，数式編集機能を持っていますが，その機能を使わなくても簡単な数式ならば文字の上付き，下付き機能やフォントの変更で数式表現を工夫できます。上付き文字は文字を行の上のほうに，下付き文字は文字を行の下のほうに配置させるもので，次のような表現ができます。

上付き文字→ 上付き文字

下付き文字→ 下付き文字

フォントやフォントサイズ，太字，斜体，など文字の書式を変更することで効果的な表現にすることもできます。

フォント→　ポップ体

フォントサイズ→　8ポイント12ポイント

太字→　太字

斜体→　*斜体*

そこで，これらをうまく組み合わせると数式表現も可能になります。

$ax^3 + bx^2 + cx + d = 0$

数式編集機能を使用して表現すると次のようになり，上のケースとあまり差異がありません。

$$ax^3+bx^2+cx+d=0$$

このように，簡単な数式ならば数式編集機能を使うこともないでしょう。[(3)] には“上付き，下付き”が入ります。

c）コンピュータのハードウェアが電気製品であり，ソフトウェアが人為的に作成されたものである以上，ハードウェアもソフトウェアも完璧なものはなく，欠陥は付きものといえます。たとえ個々の部品や機能が正常だとしても，それらが統合され，さまざまな環境で使われると，予期しない現象が起きることがあります。もともと潜在していた不具合（バグ）が表面化することもあります。例えば，ソフトウェアを使用中に突然終了してしまう，応答がなくなる，予期した結果が返ってこない，出力が正しく印字されない，などがあります。その際も，画面に何かのメッセージが出る場合と出ない場合があり，メッセージが出たとしても的確でないものもよく見受けられます。アプリケーションソフトウェアのレベルで誤りが検出できないと，OSのレベルでのメッセージになるため，抽象的で分かりにくいものが表示されてしまうからです。不具合を発見したら面倒でもメーカ又は発売元に問い合わせましょう。メーカの知らない不具合の場合には製品の修正版が出され，他のユーザの利益にも繋がります。[(4)] には“欠陥”が，[(5)] には“オペレーティングシステム”が，[(6)] には“利用者”が，[(7)] には“発売元”が入ります。

d）コンピュータを使用していて動作がうまく行かないときに，原因の特定が困難な場合があります。そうした場合，昔のコンピュータならばハードウェアもソフトウェアも同じメーカが作っていたので，そのメーカに問い合わせれば済みました。しかし今ではハードウェア，OS，アプリケーションソフトウェアとそれぞれが異なるメーカであることがほとんどです。そうなると，問い合わせる前に不具合箇所の切り分けをしなくてはなりません。ハードウェアの販売業者はハードウェアとインストール済みのOSについては保証するものの，ユーザがOSのバージョンアップをした場合は保証の対象外になってしまいます。しかし，販売時の状態に戻すには手間がかかるだけでなく，購入後の設定変更内容やソフトウェアのインストール実績などが失われることもあり，十分注意する必要があります。[(8)] には“応用プログラム”が，[(9)] には“販売時”が入ります。

e）ソフトウェアの動作環境として，主にOSの種類，CPUの能力，メモリ（主記憶）容量，ハードディスク容量が求められます。通常，製品のパッケージに書かれているので購入前に確認できます。ハードディスク容量はソフトウェアの最小構成をインストールする場合，標準的な構成をインストールする場合，すべての機能をインストールする場合で必要とされる容量が異なります。しかし，問題文にもあるように，最近市販されているパソコンに内蔵されたハードディスク容量は100GB前後の大容量です。ハードディスクに関してあまり心配することはないでしょう。[(10)] には“ハードディスク”が入ります。

例題１の解答例

(1)	(2)	(3)	(4)	(5)
14	04	06	11	07

(6)	(7)	(8)	(9)	(10)
15	16	01	09	02

例題 2　次の(1)～(10)の各記述の正誤を判断し，該当する字句を解答群の中から選べ。

(1) 記事，画像データなどの著作物の全部又は一部をそのまま抜き出して利用することを複製・転写と呼ぶ。これらを使用するためには，著作権者に転載の許可を得る必要がある。許可を得ずに転載した場合には，著作権侵害で訴訟問題になる可能性があるので，注意を要する。

(2) 私の会社では，場所の有効活用のために，CRT型ディスプレイ装置から液晶ディスプレイ装置に変更することにした。これまで使っていたCRT型ディスプレイ装置は，“資源の有効な利用の促進に関する法律（改正リサイクル法）”の対象であり，回収・再資源化の対象となる。

(3) 購入した音楽用CDに入っていた曲をMP3形式に変換し，CD-R/RW装置を使って，CD-ROMに記録し，友人達に配った。

(4) Webページの中には，アンケートの収集，懸賞への応募などで，住所，電話番号などを記入させるページがある。これらのページの中には，ダイレクトメールなどの名簿を作るための情報を集めるページ，情報を悪用するためのページなどもあり，こうしたページに気軽に個人情報を書き込んでしまわないように，十分注意する必要がある。

(5) 電子メールはネットワーク上を，通常は暗号化されずに転送されるので，秘匿性のある通信手段であるとは言えない。

(6) 短い内容をメールで伝えたい場合には，件名（Subject）に何かを書くと，本文と重複してしまうので，件名としては何も書かないようにしている。

(7) 私は，PHS受信機をコンピュータと接続し，ネットワークを利用している。受信圏外で，通話ができない場所でも，データの送受信なら可能である。

(8) ネットワーク経由で，無料のソフトウェアを入手した。このようなソフトウェアは，コンピュータウイルスに感染している可能性があるので，ウイルス検査をする必要がある。

(9) この内容は，友人のBさんから関係者に電子メールで送るべき内容であるが，Bさんはまだそのことに気づいていないようなので，Bさんの署名で私がメールを出してあげた。

(10) 一般にシェアウェアと呼ばれているソフトウェアを継続して使用している。私は学生であり，収入はないが，料金を支払う必要はある。

解答群

01　正しい又はすべきである　　　　02　誤っている又はすべきでない

［第38回(2002年7月実施)パソコン認定試験４級より］

例題２の解説

パソコンやネットワークを使うときに知っておくべき知識に関する出題です。

(1) 著作物とは創作的活動の成果物です。記事，画像データだけでなく，プログラムもホームページも電子メールなどもすべて著作物です。著作権者の許可なく著作物の内容を複製して利用することは，著作権侵害となります。よく「無断転載を禁ず」と明記されていることがありますが，たとえ書かれていなくてもしてはいけません。これに対して，「引用」は著作権法でも認められた行為です。引用の条件を満たしていれば，自分の著作物に他人の著作物を引用して利用することができます。これは“正しい又はすべきである”を選択します。

(2) 2001年4月1日の「資源の有効な利用の促進に関する法律（改正リサイクル法）」施行により，事業系パソコンの回収・再資源化の実施がパソコンメーカに義務づけられました。パソコン本体だけでなく，ディスプレイ装置も対象となります。これは“正しい又はすべきである”を選択します。

(3) MP3はMPEG-1 Audio Layer3の略で，音声圧縮の規格の一つです。音質をほとんど損なうことなく音声データを約10分の1に圧縮できるので，音楽CDの曲をMP3に変換してインターネット上で音楽の配信や配布を行うことや，オリジナルCDを作成するなどのケースが多いといわれています。音楽も著作物です。個人で楽しむ範囲での複製は認められていますが，勝手に複製を作って他で使うことや，営利目的で使用することはできません。明らかに著作権侵害です。これは“誤っている又はすべきでない”を選びます。

(4) 現在，懸賞の応募サイトは数多く存在します。また，アンケートや見積りに応じると何らかの特典を用意してあるサイトも多く見受けられます。いずれも，個人情報を記入して申込むようになっています。中にはすべての項目に回答しないと先に進めない仕組みのものもあります。こうして集められた個人情報が正しく管理されている保証はありません。商品案内などのダイレクトメールに利用されることも，他に転売されることもあるかもしれません。個人のサイトや組織の不明確なサイトにうかつに個人情報を公開すべきではありません。これは“正しい又はすべきである”を選択します。

(5) 暗号化機能を備えたメールソフトウェアもありますが，通常，電子メールは暗号化されて発信されないので，秘匿性のある通信とはいえません。したがって，パスワードや個人情報などを電子メールで送ることは避けるようにします。これは“正しい又はすべきである”を選択します。

(6) 件名を書かなくてもメールの送信をすることは可能です。しかし，件名のないメールを受取った人は，そのメールを開けてみるまで用件が分かりません。ネチケット（ネットワークを利用する上でのエチケット）にも『メールには，メッセージの内容を反映するサブジェクト（件名）ヘッダをつけなければいけません。』と明記されています。これは“誤っている又はすべきでない”を選択します。

(7) 携帯電話やPHSによるネットワーク接続では，音声通信と同様，受信圏外では通信できません。電話がかけられる受信圏内に移動してから使用するようにします。これは“誤っている又はすべきでない”を選択します。

(8) コンピュータウイルスは，ネットワーク経由で感染するケースが非常に多く，感染経路として電子メールの添付ファイルと，ダウンロードしたファイルを挙げることができます。そのために，ウイルス駆除（ワクチン）ソフトウェアを使って感染の防止を行います。しかし，コンピュータウイルスは毎日のように新種が登場するので，ウイルス駆除ソフトウェアも定期的に更新しておかなければなりません。これは“正しい又はすべきである”を選択します。

(9) たとえ善意のつもりであっても友人の署名でメールを出すということは，他人になりすますことです。決して行ってはいけません。Bさんが気づいていないようなら，Bさんに知らせてあげましょう。これは“誤っている又はすべきでない”を選択します。

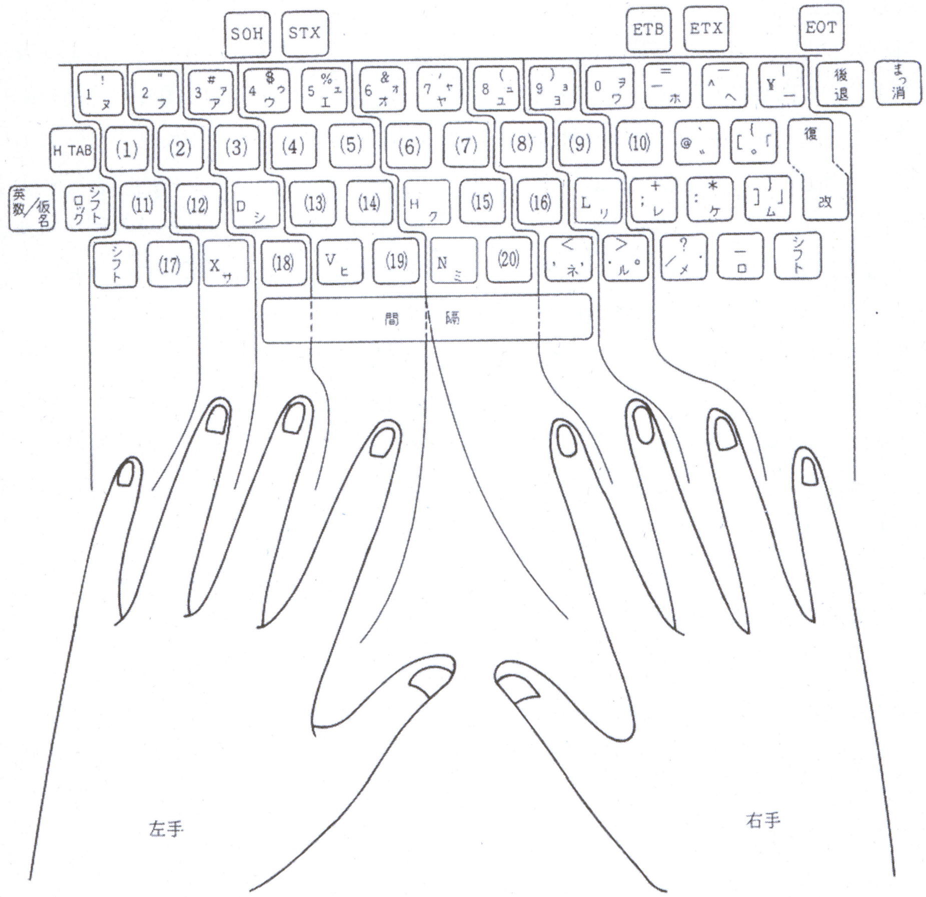
(10) シェアウェアは，実際に使って気に入れば料金を支払い継続使用するソフトウェアです。シェアウェアの中には，料金を支払うまでは機能を制限してあるもの，特定の期間が過ぎると起動できなくなるもの，などがあります。しかし，ソフトウェアを購入前に試用できるという点で，市販ソフトウェアにはないメリットがあります。これは“正しい又はすべきである”を選択します。

例題２の解答例

(1)	(2)	(3)	(4)	(5)
01	01	02	01	01

(6)	(7)	(8)	(9)	(10)
02	02	01	02	01

例題 3　次のJIS X 6002情報処理系けん盤(キーボード)配列において，(１)～(20)までのキーの英数モードの場合の文字を解答群の中から選べ。



解答群

01　A	02　B	03　C	04　D
05　E	06　F	07　G	08　H
09　I	10　J	11　K	12　L
13　M	14　N	15　O	16　P
17　Q	18　R	19　S	20　T
21　U	22　V	23　W	24　X
25　Y	26　Z		

[第38回(2002年7月実施)パソコン認定試験４級より]

例題３の解説

キーボードの配列に関する問題です。毎回必ず最後の問題として出題されるので，確実に解答できるようにしておきましょう。

1）キーの配列

特別の目的に使用する機能キーを除いて，キーボードの数字，英字，特殊文字などの配列は決まっています。このため，どのメーカのキーボードを使っても混乱することがありません。日本では日本工業規格（JIS）で決められていますが，これは国際規格（ISO）に準じているため，世界中で共通の配列ということになります。

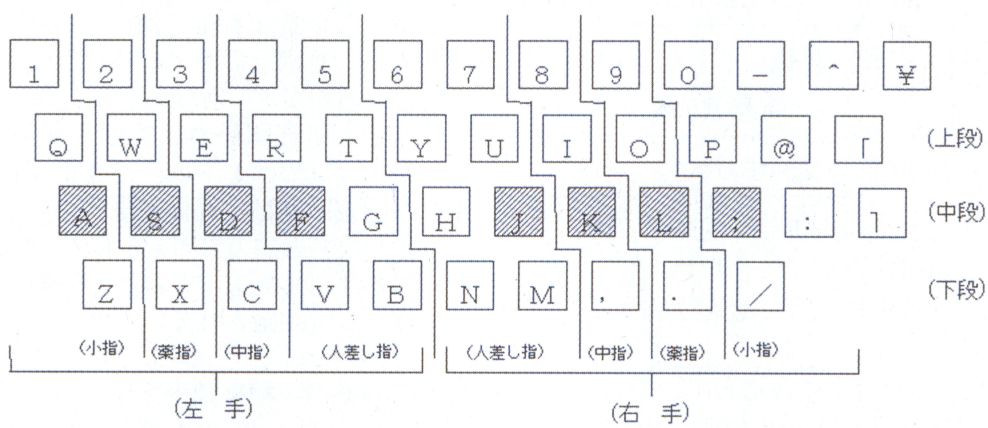
問題文にあるキーの配列は大きく横に4段の並びになっています。最上段には数字が，その下の3段には英字が配置されていますので，まずこの区別を確認しましょう。

2）ホームポジション

左右の指を置く基本位置がホームポジションです。英字キーが配置されている3段を上から上段，中段，下段と呼ぶことにすると，ホームポジションは中段にあります。右手の人差し指を[J]，左手の人差し指を[F]に置いて横にその他の指を配置していくと，右手の中指，薬指，小指は[K]，[L]，[；]のキーに，左手は[D]，[S]，[A]のキーにかぶさります。また，左右の親指はスペースキーのあたりにくるはずです。これがホームポジションです。なお，左右の人差し指を置く[F]と[J]のキーには他のキーにない突起があるので，キーを見なくてもその場所が分かるはずです。指は通常ホームポジション上に置き，必要に応じて指を上下させて，文字キーを押すようにします。

3）各指の担当するキー

指ごとの担当キーですが，だいたいホームポジションから指を上下に伸ばした帯状の範囲となります。これを次の図に示します。



各指の担当するキー

例題３の解答例

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
17	23	05	18	20	25	21	09	15	16

(11)	(12)	(13)	(14)	(15)	(16)	(17)	(18)	(19)	(20)
01	19	06	07	10	11	26	03	02	13

解説：パソコン認定試験研究会

文部科学省 大学入学資格検定 受検科目（情報関係基礎） 免除試験

第40回 パソコン認定試験

経済産業省認可公益法人・日本学術会議登録学術研究団体

社団法人パーソナルコンピュータユーザ利用技術協会

- **実施区分　3級・4級**
- **試験日　平成15年7月13日(日)**
- **願書受付期間　平成15年4月1日～5月20日**
- **合格発表　平成15年9月1日予定**

■ 試験の時間

区分	1時限（4級）	2時限（3級）
4　級	10:00～12:00	――――
3・4級	10:00～12:00	13:00～15:00
3　級	――――	13:00～15:00

■ 受験手数料（消費税込み）

区分	4級	3級	3・4級
受験手数料	3,600円	4,100円	7,200円

■ 今回実施する区分と受験資格

区分	受験資格
4　級	4級だけを受験する人。
3・4級	4級と3級を併せて受験する人（4級に合格していない人）。同日中に4級と3級を受験することができます。
3　級	すでに4級に合格している人。第36回(平成13年7月)以降に3級部分合格している人。

■ 参考：前回試験実施都市

札幌，網走，深川，旭川，美唄，青森，八戸，仙台，秋田，山形，天童，郡山，水戸，土浦，足利，さいたま，坂戸，熊谷，松戸，市川，木更津，市原，東京，吉祥寺，横浜，川崎，小千谷，富山，金沢，山梨県竜王町，長野，上伊那郡，茅野，岐阜，沼津，名古屋，豊橋，犬山，津，四日市，京都，福知山，大阪，堺，貝塚，東大阪，神戸，姫路，米子，鳥取，岡山，広島，呉，徳山，下関，徳島，阿南，高松，さぬき，松山，新居浜，高知，福岡，飯塚，長崎，諫早，佐世保，熊本，都城，那覇，宜野湾，沖縄

いま，時代はITそして情報リテラシ

“パソコンできます。”と言っても，本当の実力がどのくらいか認めてもらえません。そのような技術力を認めさせるために，一度，いま注目されているパソコン認定試験を受験してみませんか。

1983年に始まり，受験者総数は，すでに58万人，合格者総数も32万人を越えました。今では企業の採用試験の目安として使われるなど，時代のニーズにあった資格です。試験レベルは，4段階，一般的な事務処理なら3級・4級の資格で十分と言えます。

特に，パソコン認定試験合格者は，文部科学省が実施する大学入学資格検定の受検科目(情報関係基礎)が免除されます。また，学生またはキャリアアップしたい社会人にまで，パソコンの利用技術の資格試験に挑戦する傾向が急速に高まり，企業側も新規採用の条件に，また職員の昇格，奨励金の面で有資格者に優遇処置を図るところが多く，評価の対象となっています。

情報処理関係の試験はさまざまありますが，パソコンのエンドユーザを対象にした当パソコン認定試験は，経済産業省の指導のもとに，20年の実績と主要都市140都市，220会場で7月と12月の年2回実施しており，信頼性が高い認定試験として評価されています。

受験案内・願書請求方法

受験案内・願書をご希望の方は，電話，FAX，E-mail，または次のWebページからお申込みください（無料）。

http://www.pcua.or.jp/basic/

お問合せ先

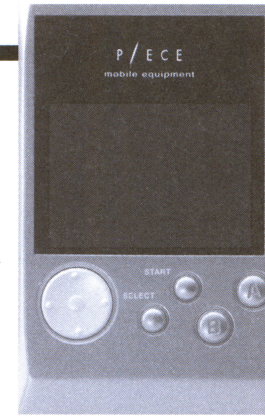
〒105-0011　東京都港区芝公園3-5-22
(社)パーソナルコンピュータユーザ利用技術協会
TEL.(03)3438-1869
FAX.(03)5472-5338
E-mail office@mb.pcua.or.jp
ホームページ　http://www.pcua.or.jp/

オリジナルソフトが作れる！
持ち歩ける！PC用携帯端末

「PIECE」プログラミング

第10回 「PIECE」情報いろいろ

文：まかべひろし



101×65×17mm（縦×横×厚さ），92g（電池含む）
発売元：アクアプラス
価格：9800円（税別）
色：シルバー（初回限定），クリアブルー，クリアレッド，パールホワイト
http://www.piece-me.com/

スペック

CPU	EPSON S1C33209（32ビットRISC，24MHzで動作）
メモリ	SRAM 256Kbyte フラッシュRAM 512Kbyte
液晶	FSTN 白黒4階調 128×88ドット
音声出力	本体出力：モノラルスピーカー 外部出力：ヘッドホン端子
インターフェース	USB／赤外線
付属フォント	JIS第一・二水準漢字フォント(10×10ドット)
電源	単三電池×1
パソコン側環境	WindowsXP/2000/Me/98（要USB）

今回は

わざわざこの記事で書くまでもない気もしますが，とりあえずの一区切り。総括的な意味も含めて，「ベーマガ最終号だから記念に買った」という人も視野にいれつつ，PIECEに関する情報をいろいろ紹介していきたいと思います。今までPIECEに興味のなかった人も，ぜひ「プログラムを持ち歩ける楽しさ」を知って下さい！

PIECEプログラミングの魅力

発売から１年経ち，PIECEプログラムの環境も整いました。そこで今一度，初心に立ち戻り，PIECEプログラムの魅力に迫ってみたいと思います。

・値段が安い（定価9,800円）

プログラムを動かすPIECE本体に加え，C言語のコンパイラやさまざまなソフトウェアがついて，1万円を切っているというのは，開発環境としては破格の値段です。USBポートがあるWindowsパソコンさえあれば，あとはPIECE本体以外の投資は必要ないのです。

・いつでもどこでも友達に見せられる

PIECE本体は非常に小さく，ポケットに入れても全然かさばらないサイズ。さっと取り出して遊ぶのも簡単ならば，自作ゲームを友達に見せるのも簡単。学校の通学時に，渾身の一作を見せるのも楽々です。

・把握しやすいスペック

今のパソコンと比べると，PIECEのスペックはチープに見えます。だからこそ，自分一人ですべてを把握しやすく，メモリの隅々まで自分でコントロールできます。また，このチープな液晶画面なら，プログラマーでも気軽に絵を描くことができるのです。

・プログラムの勉強にもってこい

使われている言語は，ゲーム開発でプロも一番使う「C言語」。プログラムの方式は，Windowsプログラムにも通じる「プロシージャ型」。もちろん，PIECEで本格的なゲームを作ることは可能ですから，PIECEプログラミングからプロのゲームプログラマーになる道もある！・・かも。

・参考資料が山のようにある

PIECE用アプリとして公開されているもののほとんどは，ソース（プログラム・リスト）公開。カーネルと呼ばれる，PIECEのOS部分まで用意されているのです。また，PIECE専門書籍「P/ECE HANDBOOK」も2冊が好評発売中。他のユーザーさんや，プロの人のプログラム・リストを見る，またとないチャンス！

携帯端末やC言語のプログラムに興味のある人，Windowsなどのプログラムが大きくて手に負えない人，趣味でプログラム開発をしたい人などにお勧めできる唯一無二の環境，それがPIECEなのです。

PIECEプログラムの手法

Windowsでのプログラムに，VisualBasic，C言語，JAVA，HSP等の開発手段が複数あるように，PIECE上で動くプログラムを作るにも，複数の手段が用意されています。

・APIプログラム

PIECEプログラミングで標準となっている方法です。PIECEのすべての機能を使うことができ，APIと呼ばれる命令セットを使ってプログラミングします。プログラムの流れは，一定時間に一度，pceAppProc()が呼ばれる「プロシージャ型」。本格的にプログラミングを楽しみたい人向け。（難易度：★★★☆☆）

・シンプルライブラリ

より簡単に，より手軽にプログラムができるとして，本体発売後のアップデートで公開された方法です。BASICのような上から下へ流れるプログラムを記述でき，スプライトの描画に至ってはAPIプログラムの画像描画より早いという優れもの。機能は制限されていますが，簡単にPIECE上で動くものを作ってみたい人向けです。（難易度：★★☆☆☆）

・P/ECE VISUALNOVEL SYSTEM

http://zura.chu.ath.cx/piece/pvns.html

ヅラChu氏が制作・公開されている，PIECE用アドベンチャーゲーム（ADV）

ツールです。単純にADVを作りたい場合，非常に簡単にプログラムと絵素材だけで作成することができます。自分の絵を使ってPIECEゲームを作ってみたい人は，まずここから始めてみるのが良いのではないでしょうか。(難易度：★☆☆☆☆)

その他，PIECE上でJAVAアプリを動作させたり，BASICを動かしたりするソフトなども公開されています。自分の気に入った開発環境で作っていき，物足りなくなったら別の環境に移る，くらいの浮気性であってもいいかもしれません。

ちなみにボクのオススメは，シンプルライブラリでの開発。アルゴリズムの動作検証や，簡単なプログラムを行うのなら，シンプルライブラリは手放せません。

シンプルライブラリでの作り方

シンプルライブラリは，書式的にはC言語に乗っ取ったものですが，限りなくBASICに近い感覚で，手軽に作ることができる開発環境です。どれくらい簡単かと言うと，画面にメッセージを表示するプログラムは，以下のようになります。

```
printstr( "Welcome to PIECE World" );
wait( 10000 );
```

文字を画面に出して，10秒待つというプログラム。こんなシンプルな，直感的なプログラムが書けるのです。しかし，簡単すぎると侮ってはいけません。工夫次第で，パックマンのようなドットイートゲームや，シューティングゲーム，はたまたパズルゲームのように複雑なものも作ることができるのです。また，C言語に慣れた人なら，次のように記述することも可能です。

```
// simplelibを使う為に必要
#include "simple.h"

// 変数定義(simplelib標準)
int a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,
u,v,w,x,y,z;

void smain() {
    printstr( "Welcome to PIECE World" );
    wait( 10000 );
}
```

この形なら，サブルーチンを関数にまとめたり，配列や構造体を自由に使うことができます。感覚は，ほぼ，DOSでのプログラムと同じですので，昔プログラムを組んでいた人も，すんなり入ることができるのではないでしょうか。ポイントは，void smain() が起動関数であること，ただこれだけです。

APIプログラムでの作り方

APIでのプログラミングは，PIECEのすべての機能を引き出すことができる開発環境です。アプリの多くはこの環境で作成されているため，サンプルやTipsも多く，作り方さえわかってしまえばPIECEを好きなように操ることができます。シンプルライブラリでは，大きな絵を扱うことが難しいのですが，APIプログラムでなら可能です。その他，他環境で制限されていることが，こちらでは制限されていない，ということが一番のメリットでしょう。

ただ，APIプログラムを作成するには，その挙動を理解しなくてはいけません。通常のC言語ではmain()がプログラムのスタート位置，シンプルライブラリではプログラムの一番頭もしくはsmain()がプログラムのスタート位置になっていますが，APIプログラムでは，

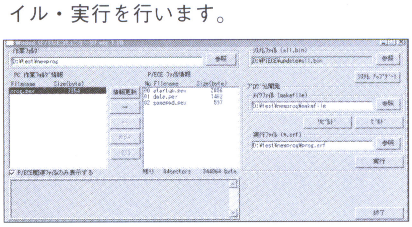
- 起動関数 pceAppInit()
- 実働関数 pceAppProc()
- 終了関数 pceAppExit()

という三つの関数を用意し，それぞれに対応したプログラムを正しく記述しなく

シンプルライブラリでのプログラム準備

①作業場所となるフォルダを作る
エクスプローラなどで，フォルダを作ってください。
②元となるファイルをコピーしてくる
(PIECEをインストールしたフォルダ)¥app¥simple_hello から，
- makefile
- prog.c
- pat.c
- pat.bmp
- prog.txt

の五つのファイルをコピーします。
この中のprog.txt，もしくはprog.cを書き換えていきます。作り終わったら，PIECEコミュニケータ（WinIsd）を使ってコンパイル・実行を行います。



①PIECEコミュニケータ（WinIsd）を起動し，設定を行う
「プログラム開発」の「メイクファイル(makefile)」に，コピーした先のmakefileを指定してください。
②「リビルド」ボタンを押す
このボタンを押すと，ソースファイルを元に，実行ファイルを作ります。
③実行ファイルの設定をする
「実行ファイル（*.srf)」に，できあがったprog.srfを指定します。
④PIECE本体とPCが接続されているか確認する
実行は，PIECE本体上で行われます。ちゃんとPCとPIECEが接続されているか，確認してください。
⑤「実行」ボタンを押す
作ったプログラムをPIECEに転送し，実行します。
動いたでしょうか？この手順は，プログラムや画像を書き換える前に，あらかじめ確認しておくと良いかもしれません。最後に，できあがったプログラムを持ち歩くためにmakefileを書き換えます。

```
$(PRGNAME).srf : $(OBJS)
        $(LD)     $(LDFLAGS)
e$(PRGNAME)
.srf $(OBJS)
```

この下に，1行追加します。

```
$(PRGNAME).srf : $(OBJS)
        $(LD)     $(LDFLAGS)     -
e$(PRGNAME).srf $(OBJS)
        ppack -e $(PRGNAME).srf -
o$(FILENAME).pex -n$(CAPTION)
```

これを行ったあと「リビルド」しますと，prog.pexが作成されます。後は，他のアプリケーション同様，PIECEコミュニケータでこのファイルをPIECEに転送すれば，プログラムを持ち歩くことが可能になります。
命令などの資料は,(PIECEのソフトをインストールしたフォルダ）¥docs¥SIMPLE¥index.html にあります。

てはいけません。まず，以下のソースを見てください。これはまったく何もやらない，PIECE最小のプログラムです。

```
#include <stdio.h>
#include <piece.h>

void pceAppInit( void ) {
 pceAppSetProcPeriod( 80 );
}

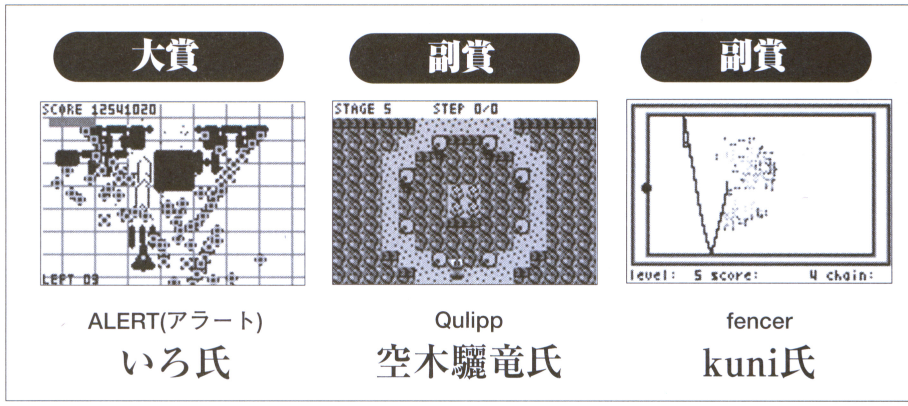
void pceAppProc( int cnt ) {
}


void pceAppExit( void ) {
}
```

最初に実行されるのはpceAppInit()です。ここで指定している pceAppSet ProcPeriod(80) は，「pceAppInit()を抜けたあと，pceAppProc()を80/1000秒間隔で呼び出すこと」という内容です。で，プログラムのメインはpceAppProc()で記述するというのがお約束事になっています。

PIECEの動作は本連載で再三出てきていますので(2002年11月号117ページ)参照してください。

この図を見ると，アプリの終了処理や設定メニューなどは，pceAppProc()が呼ばれる合間に処理されることがわかります。また，その他パッドや音の処理なども行われているので，pceAppProc()の中でループを行ったりするのはご法度であるとわかるでしょう。このように，定期的にOS側から呼ばれる関数をプロシージャと呼びます。プロシージャ型プログラムは，別にPIECEに限ったことではありません。WindowsやMacOSで動くプログラムも，同様の考え方を元に組まなくてはいけません。逆に言えば，PIECEでプロシージャ型に慣れてしまえば，需要の大きいWindowsプログラムを勉強するときも，スムーズにいくと言えるでしょう。BASICやHSPなど，対話型言語にはない考え方ですが，覚えておいて損はありません。



《第1図》アクアプラス公認コンテスト入賞作

新カーネル制作中！？

記事執筆時にはまだ公開されていませんが，本誌独自の情報網によると，1年近く更新の無かったPIECEのカーネルですが，v1.20（？）が制作中とのこと！　内容は機能追加よりも，バグ修正や，多くのユーザさんが見つけてきた問題解決が盛り込まれるとのこと。個人的には，以前本連載で紹介した，待ち受け時の省電力化がカーネル側で入ると，多くの人がPIECEを持ち歩けるようになるのではないかと，期待しています。

アクアプラス公認 コンテスト速報！

先日，PIECEの公式Webサイト(http://www.piece-me.com/)にて募集していたコンテストですが，ついに結果が発表されました！(**第1図**)

入賞作をはじめ多くの応募があり，かなりの盛り上がりを見せていました。次のコンテストも楽しみなところ。今回応募しなかった人も，次のコンテストまで作品を仕込んでおこうっ！

PIECEゲーム収録！ 「テネレッツァ」発売

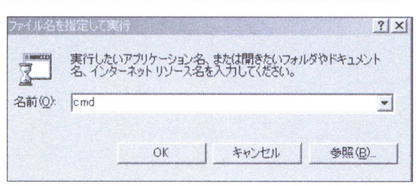
この冬，XBOX版が発売された3DアクションRPG「テネレッツァ」が，Windows版にパワーアップして登場！　暖かみのある色彩のポリゴンと，派手な魔法エフェクト，ノリの良いシナリオが売りの本ソフトですが，PIECE制作元のアクアプラスが開発しただけあって，Windows版ではPIECE対応ソフトを収録！　その名は「ロロ助危機一髪」。タ

APIプログラムでのプログラム準備

①作業場所となるフォルダを作る
　エクスプローラなどで，フォルダを作ってください。今回は ¥PieceProgram¥apiprog¥ とします。

②元となるファイルをコピーしてくる
（PIECEをインストールしたフォルダ）¥app¥hello から，
　・makefile／・hello.c
の二つのファイルをコピーします。この中のhello.cを書き換え，プログラムを作成します。

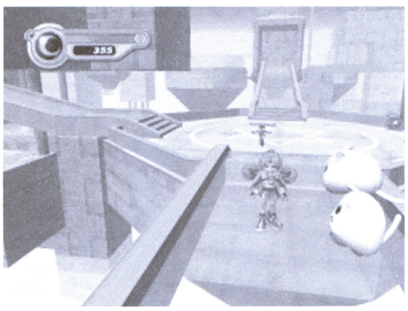
●**コンパイル**
①コンソールを開いてカレントディレクトリを移動する
　コンパイルは，コンソールウィンドウを用いて行います。Windowsのスタートメニューから，「ファイル名を指定して実行」を選び，
　・Windows98/Meの場合は「command」
　・Windows2000/XPの場合は「cmd」



と打ち込み，実行します。コンソールウィンドウ（DOS窓）が開いたら，
　cd ¥PieceProgram¥apiprog
と打ち込み，カレントディレクトリをプログラムフォルダに移動します。
②コマンドをそれぞれ打ち込む
　コンパイル（実行ファイルを作る）を行う時は[make]，作った実行ファイルをPIECE上に転送して動かすには[run]，PIECEにWinlsdで転送できるpexファイルを作るには[make pex]とそれぞれ打ち込みます。
　命令などの資料は，PIECEのソフトをインストールしたフォルダ¥docs¥API¥index.html にあります。



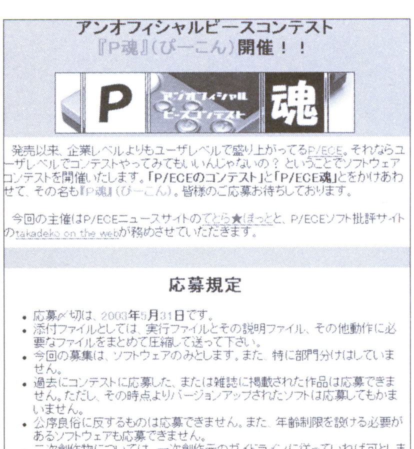
《第2図》「テネレッツァ」のゲーム画面


《第3図》「テネレッツァ」のゲーム画面

イトルから，シビアなアクションゲームであることが予想されるが・・？　気になる人は，要チェック！(第2,3図)
※3Dアクションゲームということもあり，パソコン側へのスペック要求も高くなっています。「テネ・ベンチ」という動作検証用ソフトが公開されていますので，あらかじめこのソフトでチェックしておくことをオススメします。
http://www.aquaplus.co.jp/

ユーザ主導のコンテスト発動！

常にユーザー主導で動いてきた携帯端末PIECE，ついにユーザー主導でのソフトウェアコンテストが開催されることとなりました！その名も「P魂（ぴーこん)」。



《第4図》『P魂』の募集画面

熱い思いと，なんとなく懐かしい響きのこのコンテスト，主催者がユーザーなら，盛り上げていくのももちろんみんなだ！応募〆切は2003年5月31日。個人で，学校のクラブ活動で，どんどん作品を作って応募しようっ！(第4図)
http://zura.chu.ath.cx/pkon/

現在稼働中，オススメWebサイト

PIECEのWebサイトは数あれど，とりあえずここをチェックしておかなきゃはじまらない。そんな珠玉のWebサイトをご紹介。

てとら★ぽっと
http://zura.chu.ath.cx/

日々更新されるWeb世界で，PIECE情報を知りたいならここ。各PIECEサイトの更新情報が毎日のように掲載されています。PIECEサイトを立ち上げた時は，是非このサイトにご報告を。

P/ECE研究室
http://north.hokkai.net/~nsawa/

技術面ではピカ一。PIECEプログラムを開発するのなら，理解できなくてもここの記事はすべて見ておこう。読むだけで，プログラムスキルのレベルアップは間違いなしです。

takadeko on the web
http://www.d4.dion.ne.jp/~kennjit/

「P/ECE批評家さん」では，ユーザーさんがユーザーさんのアプリを評論しています。厳選紹介しているので，このサイトで紹介されれば，あなたも一人前のPIECE開発者ですね。

P/ECE WebIndex
http://wakufactory.mine.nu/piece/

無尽蔵に増え続けるPIECEアプリを一元管理してるのがこちら。新しいPIECEソフトを探しているなら，ここで決まりです。また，より便利にするために，PIECEソフトを発表したら，WebIndexに登録しておきましょう。

HANDBOOK Vol.1 / Vol.2も是非！

手前みそで恐縮ですが，PIECEだけを扱った単行本「P/ECE HANDBOOK」が，Vol.1，Vol.2とも好評発売中です。Vol.1は「作る」ことに重点を置いた，PIECE総合誌。Vol.2はPIECEで「遊ぶ」ための本となっています。どちらも，PIECEユーザーには必携の本との評価をいただいております。まだ買っていない人は，お店から無くならないうちに是非お買い求め下さい！
http://www.jc2.co.jp/p/piecetop.html

また会おうねっ！

長いようで短かったPIECE連載，いかがだったでしょうか。正直，好き勝手やらせてもらいました。自分の好きなもので連載を持たせていただいて，ベーマガの懐の深さには感謝してもしきれません。熱い気持ちをそのまま記事にぶつけるライブ感，時には熱くすぎて空回りしてしまったこともあったけれど，みなさんに伝わっていれば嬉しく思います。

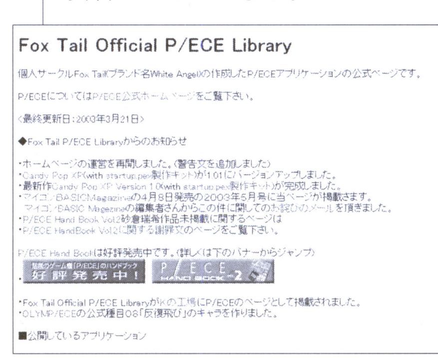
PIECEに限らず，プログラムは楽しいもの。その中でも，「いつでもどこでも友達に見せられる」PIECEプログラムは，とびきり楽しいものだと断言できます。またいつか，PIECEの熱さを伝えるためにボクは記事を書きます。近い未来，また，みなさんと会えますように。

ご愛読ありがとうございました。

今月のPIECE厳選Webサイト

Fox Tail Official P/ECE Library
http://www.punyu.net/~angeltea time/piece/

小さなP/ECEアプリケーションソフトを多数公開しているWebサイトです。単機能のものを多数用意しているので，自分で欲しい機能を選ぶことができるだけでなく，その機能の参考資料となるでしょう。





BM 基礎講座

番外編

ベーシック プログラミング教室

星音高校

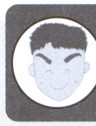
舞台は，星音(ほしね)高校パソコン研究部。創部2年目の今年，新入部員がたくさん入ってくるとの噂。先輩になる部員たちは新入部員に負けないように，１年間の総まとめをすることに。また，今年チャレンジしようと思っている新しいプログラムの研究発表もあるようです。みんながんばれ！

登場人物


「ちぃさん」(ちぃ)・・・3年生。部長。今年は新入部員がたくさん来る！という噂を聞いてから，ずっとそわそわ。部長として，いろいろと考えることがあるようです。


「みぃちゃん」(みぃ)・・・3年生。新入部員には女の子もいるらしい！？わくわくしています！


「よっさん」(よっ)・・・OB。頼りになる先輩。星音高校を卒業後，音楽スクールへ通うことに。めざすはプロドラマー！？


「ゆうちゃん」(ゆう)・・・２年生。ちょっと落ち着きがでてきました。プログラミングが上達してきた余裕から？


「しんちゃん」(しん)・・・２年生。釣りクラブの部長をこなす釣りキチ。今年からかけ持ちで，パソコン研究部に入部。よっさんに負けないゲーム好きです。

パソコン研究部～創部１周年！

パソコン研究部（部室）

みぃ：「星音高校パソコン研究部，創部1周年！！おめでとー！」

ちぃ：「ほんとだね！あっという間の1年だったね。みんなおつかれさま。」

みぃ：「今年は新人部員がたくさん来るみたいね。」

ちぃ：「うん。顧問の増田先生の話によると，部室ももっと広いところに変わるみたい。」

みぃ：「やったね！で，どこに変わるの？」

ちぃ：「それはまた後で報告します！お楽しみに～。」

みぃ：「なんかもったいぶってる(笑)」

ちぃ：「これからは，僕らが新入部員に指導していかないといけないよね。ベーシックプログラミング教室のバックナンバー(2002.4～9，2002.11～2003.4号)でしっかり復習して指導できるようにしておきましょう。」

みぃ：「ほんとね。私たちがしっかりマスターしておかないとね。」

ちぃ：「みぃちゃんとゆうちゃんで，これまでに習ったコマンドをまとめたリストを作ってみて。僕は，いま取り組んでるプログラムの研究発表をするよ。」

みぃ：「VBAだっけ？」

ちぃ：「うん。エクセルを使ってやってるんだ。そういえば，ゆうちゃんは？」

みぃ：「もうすぐ来ると思うよ。」

ちぃ：「じゃあ，みぃちゃん先に進めておいてくれるかな。」

みぃ：「うん。まかせといて。ここ狭いから，ミーティングルームに行ってます！」

ちぃ：「がんばってね。よし，じゃあ僕はVBAのレポートを・・・」

～ガラガラ（部室の戸が開く）

ゆう：「こんにちは～！遅くなってごめんなさい。」

ちぃ：「あ，ゆうちゃん待ってたよ。お！そこにいるのは，しんちゃん？」

しん：「こんにちは。」

ちぃ：「久しぶりの登場だね。だけど，タイトルにはいつも出ていたね。」

ゆう：「そういえば……星音高校の文字の上で音符を釣ってる(笑)」

ちぃ：「釣りといえば，釣りクラブの部長さんがどうして，ここへ？」

ゆう：「僕が誘ったんだ。パソコン研究部に入部しない？って。」

しん：「釣りクラブと，かけ持ちになりますがいいですか？」

ちぃ：「いいよ！ゆうちゃんと同じ学年の部員いないから，ゆうちゃんも寂しかったと思うしね。」

しん：「ありがとうございます！！」

ちぃ：「がんばってね。まあ今日は見学ということで，ゆっくり見て行ってよ。ゲームもあるしのんびりしていって。」

しん：「はい！」

ちぃ：「ゆうちゃん，早速だけど，みぃちゃんを手伝って欲しいんだ。」

ゆう：「はい。手伝う？」

ちぃ：「詳しくは，ミーティングルームでみぃちゃんに聞いて。」

ゆう：「はい！わかりましたー！行ってきます～」



(2時間後)

みぃ：「ふー。やっとできたー。けっこう時間かかったねぇ。」

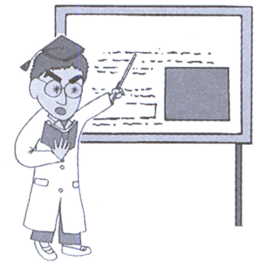
ゆう：「ほんとですねー。さぁ部長に

見せに行きましょう！」
みぃ：「そうね。」
ゆう：「部長！できました。」
みぃ：「これまでに勉強してきた主なベーシックコマンドを集つめてみました。」

ちぃ：「がんばったねー」
みぃ：「うん。結構たくさんあったね。」
ゆう：「これまでに習ったものを表にしてみたので（**下表**），忘れたときにすぐに見つけることができます。」
ちぃ：「ほんとだね。この表のあるものは基本だからしっかり覚えておいてね。」
ゆう：「バックナンバーもしっかり揃えておかないとね。」



みぃ：「部長のほうはできた？」
ちぃ：「うん，ばっちりだよ。今回はエクセル2000のVBAを使ってプログラムしてみます！ベーシックをある程度マスターできたら，みんなもチャレンジしてみよう！」
ちぃ：「VBAとは，Microsoft社のVisualBasicの仕様に基いたプログラムです。Officeシリーズに搭載されているので，エクセル，ワード，パワーポイント，アクセスなどでも使用できます。」
みぃ：「へーそうなんだー」
ちぃ：「VBAは，VisualBasic for Applicationsの略です。エクセルで作ったプログラムはエクセルがインストールされていれば動作します。ということで早速やってみましょう！」
ちぃ：「VBAではマクロとして自動記録できる機能と，VisualBasicEditor（VBエディター）を利用してマクロの編集する機能があります。ここで



キーボード関連

INKEY$（関数）	キーボードから1文字入力	例）A$=INKEY$
INPUT	キーボードから文字の入力	例）INPUT "名前を入力してください";A$

プログラム制御

END	プログラムの終了	例）END
FOR～NEXT	繰り返し実行	例）FOR I=0 TO 7:PRINT I:NEXT I
GOSUB～RETURN	サブルーチンへ分岐　RETURNでサブルーチン終了	例）GOSUB *key
GOTO	強制分岐	例）GOTO 200
IF～THEN～ELSE	条件による分岐	例）IF A=1 THEN PRINT "OK" ELSE PRINT "NO"
ON ERROR GOTO	エラー処理	例）ON ERROR GOTO *ERRSYORI
ON～GOTO(GOSUB)	条件による分岐（サブルーチンへ分岐）	例）ON I GOTO 100,300，*submode,*keycheck
REM(')	注釈	例）REM プログラム制作開始　2003/4/8

画面

CIRCLE	円を描く	例）CIRCLE(400,200),50,7,,,,F,2
CLS (0,1,2)	画面のクリア	例）CLS 2
COLOR	表示カラーの設定	例）COLOR 5:PRINT "テスト"
LINE	線，ボックスの描画	例）LINE (100,100)-(200,200),5,B
LOCATE	カーソル位置の設定	例）LOCATE 10,10:PRINT "テスト"
PAINT	グラッフィク領域の塗りつぶし	例）PAINT (200,200),1,7
PRINT	文字の表示	例）PRINT "表示"
PSET	点の表示	例）PSET(100,100),6
BLOAD	ビットマップファイルをメモリに読み込み	例）BLOAD "TEST.BMP",A(0)
PUT@	メモリに読み込んだビットマップファイルを表示	例）PUT@(100,100),A(0),PSET

ファイル

OPEN	ファイルのオープン	例）OPEN "TEST.TXT" FOR OUTPUT AS #1
PRINT #	ファイルへ書き込み	例）PRINT #1,"BASIC"
CLOSE #	オープンしているファイルを閉じる	例）CLOSE #1
INPUT #	ファイルから読み込み	例）INPUT #1,A$
LINE INPUT #	ファイルから改行単位で読み込み	例）LINE INPUT #1,TMP$

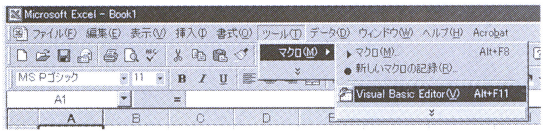
定数と変数

DATA	READ命令で読み込むデータをセット	例）DATA 100,200,"TEST"
DIM	配列の宣言	例）DIM A(10)
READ	DATA命令でセットされているデータの読み込み	例）READ A,B,C$
RND(関数)	乱数の発生	例）A=RND(1)
RANDOMIZE	乱数発生の初期化	例）　RANDOMIZE
SWAP	変数の中身を入れ替え	例)SWAP A,B

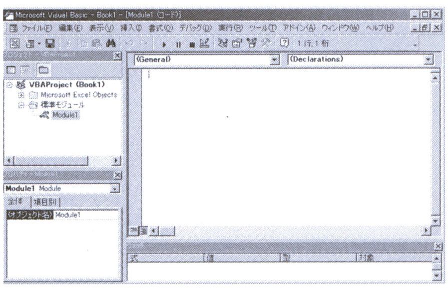
関数

DATE$	現在の日付を取得
TIME$	現在の時刻を取得
CHR$	アスキーコードに対応した文字へ変換
INSTR	文字列の中から指定された文字列を探してその開始位置を取得
LEFT$	文字列の左端から指定した長さの文字列を切り出す
LEN	文字列の長さを取得
MID $	文字列の中の任意の位置から指定した長さの文字列を切り出す
RIGHT$	文字列の右端から指定した長さの文字列を切り出す
SPACE$	指定した数の空白文字列を作成

は，VBエディターを利用してプログラミングする手順を紹介します。エクセルを起動し[ツール]-[マクロ]-[VisualBasicEditor]を起動します。」



ちぃ：「VBエディターが起動したら，[挿入]-[標準モジュール]を選び，標準モジュールを追加します。VBAProjectウインドウに標準モジュールが追加されます。」



みぃ：「左のウインドウね。」

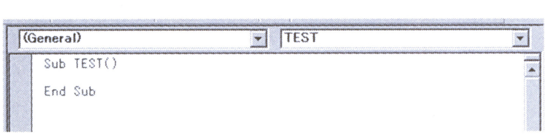
ちぃ：「もしプロジェクトウインドウが見つからない場合は，[表示]-[プロジェクト エクスプローラ]で出てきます。これで準備完了です。」

ゆう：「右の大きなウインドウにプログラムを書いていくんですね。」

ちぃ：「そうだよ。だけどこれまでと違う点が何点かあるんだ。まずプログラムに行番号は要らないんだ。行番号はなくても上から順番に実行するのは同じだよ。次に，プログラムはひとつの実行単位をSub～End Subの間に記入します。」

みぃ：「Sub～End Sub？」

ちぃ：「右のウインドウの中に，Sub TESTと入力して[ENTER]キーを押してみよう！」



ゆう：「あ，自動的にEnd Subが付加された！」

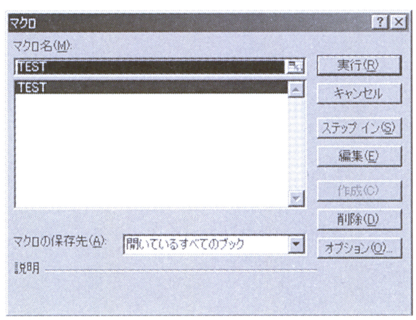
ちぃ：「この間にプログラムを書いていきます。せっかくエクセルと連動しているので，エクセルのセルに値を書き込むプログラムを作ってみよう。」

```
Sub TEST()
Cells(1,1).Value="こんにちは"
End Sub
```

ちぃ：「1行だけだけど入力してみて。入力ができたら実行します。実行方法は2種類あります。エクセルから実行する方法とVBエディター上から実行する方法です。まずエクセルから実行する方法をやってみよう。」

ゆう：「エクセルから実行するということは，このままエクセルの画面に戻ったらいいんですね。」

ちぃ：「そうです。エクセルに戻ったら[ツール]-[マクロ]－[マクロ]を実行するとマクロウインドウがでてきます。マクロウインドウには，Sub TESTで名前をつけたTESTが登録されています。」



ちぃ：「TESTを選択し実行します。するとエクセルのワークシート上に，「こんにちは」と表示されます。」

みぃ：「ほんとだ！ということは,Cells(1,1).Valueというのがエクセルのシートの値を設定する命令になるんだね。」

ちぃ：「そうだね。シンプルだからわかりやすいだろ。(1,1)というのがセルの座標です。(行,列)になっているので，C5セルの場合(5,3)になります。」

ゆう：「なるほどー」

ちぃ：「これに今まで習ったベーシックのコマンドをミックスすることで，いろいろとプログラミングできます。」

```
Sub 九九()
    Dim I As Integer
    Dim J As Integer

    For I = 1 To 9
        For J = 1 To 9
            Cells(J, I).Value = I * J
        Next J
    Next I
End Sub
```

ちぃ：「これはエクセルのシート上に九九を表示したものです。最初のDim I As Integer という表現がこれまでと違う部分だね。Dimはベーシックでは配列を設定していただけど，ここではIという変数を整数型で使用するという宣言です。」

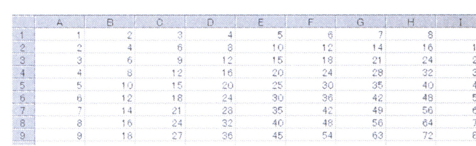
ゆう：「実行してみますね。」

ちぃ：「うん。今度はVBエディターから実行してみようか。」

ゆう：「はい。」

ちぃ：「実行したいSub ～ End Subの中にカーソルを合わせて，VBエディターの[実行]メニューかツールバー上の▶を押すと実行できます。実行したらエクセルに戻って確認してみましょう。」

ゆう：「ほんとだ！九九が表示されていますね。これそういえば，ベーシックでもやりましたね。」



	A	B	C	D	E	F	G	H	I
1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	10	12	14	16	18
3	3	6	9	12	15	18	21	24	27
4	4	8	12	16	20	24	28	32	36
5	5	10	15	20	25	30	35	40	45
6	6	12	18	24	30	36	42	48	54
7	7	14	21	28	35	42	49	56	63
8	8	16	24	32	40	48	56	64	72
9	9	18	27	36	45	54	63	72	81

ちぃ：「ほとんど変わらないよね。PRINT命令がないけどね。」

みぃ：「IF～THEN命令も使えるの？」

ちぃ：「もちろん！次のプログラムを入力してみよう！」

```
Sub 偶数()
    Dim I As Integer
    Dim J As Integer

    For I = 1 To 9
        For J = 1 To 9
            If Cells(J, I).Value Mod 2
 = 0 Then Cells(J, I).Value = "○"
        Next J
    Next I
End Sub
```

ちぃ：「さっきの九九のプログラムを実行した後に実行します。各セルの中身が偶数ならば，○印に置き換わるようになっています。実行すると,」

	A	B	C	D	E	F	G	H	I
1	1	○	3	○	5	○	7	○	9
2	○	○	○	○	○	○	○	○	○
3	3	○	9	○	15	○	21	○	27
4	○	○	○	○	○	○	○	○	○
5	5	○	15	○	25	○	35	○	45
6	○	○	○	○	○	○	○	○	○
7	7	○	21	○	35	○	49	○	63
8	○	○	○	○	○	○	○	○	○
9	9	○	27	○	45	○	63	○	81
10									

ちぃ：「となります。」
みぃ：「おもしろい！！これっていろいろと応用できそう！」
ちぃ：「うん。FOR～NEXTとIF～THENが使えることでいろいろと幅が広がってくるよね。」
ゆう：「ほかにもいろいろできるんですね。」
ちぃ：「今回は紹介できなかったけど，VBエディターの[挿入]-[ユーザーフォーム]を使うことで，フォームを使ったプログラムも作れるんだ。これはとても便利だよ。」
みぃ：「今年はこれに本格的にチャレンジしたいなぁ。」
ゆう：「僕もやってみたいなぁ。」
ちぃ：「ハハハ。みんなすっかりハマっちゃったね。でも本当に便利だよ。自動的に記録するマクロ機能と併用することで，かなりプログラムも楽だし。がんばって行こうね！」

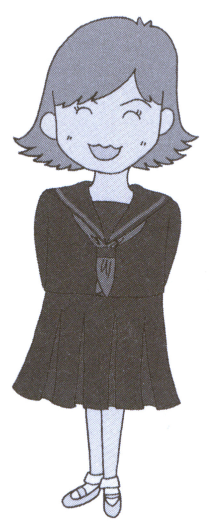
～ガラガラ（部室の戸が開く）

よっ：「よぉー元気か！」
みぃ：「あ，よっさん先輩！！」
ゆう：「こんにちは！お久しぶりです」
ちぃ：「こんにちは！先輩，ドラムの専門学校に行ってるんですよね！」
よっ：「せや。なんや，ちぃさん，もう知ってたんかいな。一応プロめざそうかなと思ってな。これ昨日撮った写真や。」



みぃ：「ほんとだ！かっこいい！」
ゆう：「だけど……なんで学生服なんですか……」
よっ：「いやなぁ。ずっとこれでやってたから，他の服じゃ落ち着かんのや。」
ちぃ：「アハハ（苦笑）」
よっ：「そうそう。さっきそこで俺のファンやゆう子が居たんで連れてきたんや。」
みぃ：「よっさん先輩のファン？」
よっ：「はよ，はいっといでや」
まい：「はい。みなさんこんにちは。」

みぃ：「あれ？まい。何やってるのこんなところで？」



一同：「まい？？」
みぃ：「私の妹なの。」
一同：「妹？？」
まい：「今日は，教科書の購入日だったので，帰りに寄ってみたんだ。」
みぃ：「よっさん先輩のファンだったの？」
まい：「うん。もう卒業しちゃったのが残念だけど……」
よっ：「人気者は照れるなぁ～。まいちゃん入部したいそうだから，よろしく頼むわ。」
まい：「よろしくお願いします。」
ちぃ：「あ，こ，こちらこそよろしくお願いします。」
みぃ：「そんなこと家では全然言ってなかったじゃない。もう……」
まい：「えへへ。」
ゆう：「わーい。楽しくなりそうだね。」
ちぃ：「そうそう。はじめに話してた部室の件だけど……」
みぃ：「うんうん。」
ちぃ：「星音高校は，http://www.hoshine-hs.net で創部2年目もがんばっていきます！」
ゆう：「わぁ。すごい！星音高校ドットネット！」
ちぃ：「これまでのベーマガでの活躍はもちろん，新しい星音高校のイベントなど盛りだくさんになる予定です。みなさんぜひアクセスしてください。」
みぃ：「うわぁ。すごいねぇ。がんばらないといけないねー！」
ちぃ：「ベーシックプログラミング講座の通信講座も続いていますので，こちらへのアクセスもお待ちしております！
http://www.beeplus.or.jp/bm/basic/」
ゆう：「ベーマガは終わってしまいますが，僕らのプログラムスピリットは永久に不滅です！！」
一同：「これからも星音高校をよろしくお願いします！！」

（記事内の星音高校および登場する人物は架空のものです）

講師紹介

川上朝幸（Kawakami Tomoyuki）
1971年8月31日　大阪府生まれ



93年カプコン入社。サウンドエンジニアとして，「ストリートファイター２」「ヴァンパイア」「ロックマン」シリーズをはじめ「バイオハザード」など，多くのサウンド制作を経て00年退職，翌01年「スタジオ スターノーツ」を起業。サウンドエンジニアとしての活動を軸にスクール講師，プログラミング，ホームページ，ビデオ制作や，カメラ撮影，アマチュアバンドのアルバムプロデュース，自身のバンド「スターノーツ」のライブなど幅広く活動している。

STARNOTES
http://www.starnotes.jp
basic@starnotes.jp

この講座で使用しているプログラムデータは下記のホームページでダウンロードすることができます。
http://www.beeplus.or.jp/bm/basic/
☆星音高校ドットネット
http://www.hoshine-hs.net
☆ＢＭ基礎講座「続！ベーシックプログラミング教室」
制作 スタジオ スターノーツ
（http://www.starnotes.jp/）
協力 有限会社 ビープラス
（http://www.beeplus.or.jp/）



でんでんタウンロボットクラブ奮戦記

第3号

今後の活動に夢ふくらむ！（最終回）

動かす！LEGOに挑戦しました



プチプチとパーツをはめこんで自由に組み合わせができるLEGOは魅力的なホビーですが，3月2日（日），でんでんタウンロボットクラブでは“動かす！LEGO”に挑戦しました。

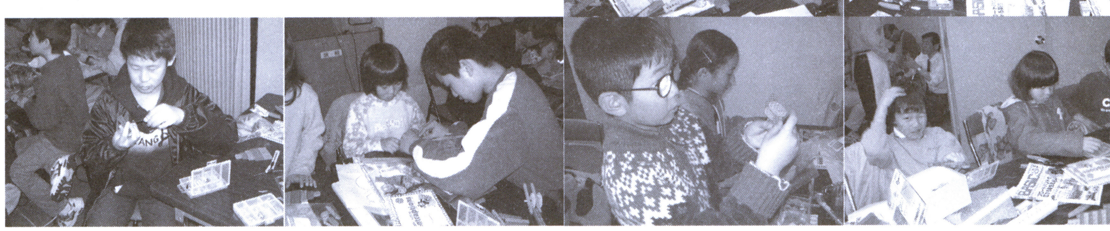
動力はマブチの130モータです。このモータは1.5Vで動くので模型工作にピッタリです。負荷がかかっている時の1分間のカタログ回転数は約6000回転ですので，1秒間に100回回ります。1秒間に1回動いたらいいロボットの足などに動力を伝える場合は，100分の1に回転を下げればクランクとして使えます。

速さが100分の1になれば力（トルク）は逆に100倍になります。今回でんでんタウンロボットクラブではLEGO純正のウオームギアを使いました。ウオームギアは歯が1と考えられるので，組み合わせる歯車の歯の数を数えれば，それがそのまま減速比になります。ウオームギアっておもしろいですね。

LEGOのウオームギアは直接マブチモータに取り付けられないので，タミヤの8枚歯のピニオンギアを4枚に削って改造して使いました。LEGOを長年いじっている川森先生のアイデアです。

そんなこんなで今回はとにかくLEGOで作った「しゃくとり虫とブルドッグ」が動きました。



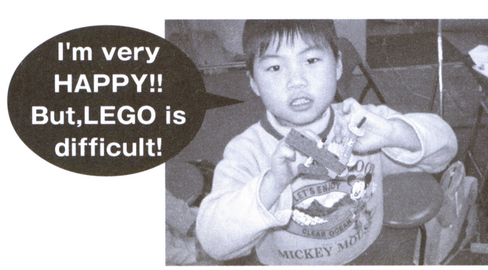


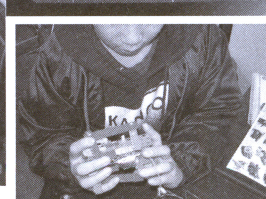
洋書LEGOクレージーの付属パーツに追加しました

教材のLEGOクレージーは洋書のため，インターネット網をもつ有名な「アマゾン」を利用して入手しましたが，パーツが60個しか入っていないので，今回あちこちに手配してパーツを追加しました。

今月紹介のしゃくとり虫と4足歩行のブルドッグはLEGOクレージーだけでは作れませんのでご注意ください。このLEGOクレージーはパソコンの翻訳ソフトで全文和訳しましたが，直訳では「いくらかの標準LEGOれんがと同様に袋または基礎板から……」など楽しい珍訳となりました。皆さんも翻訳に挑戦してね！



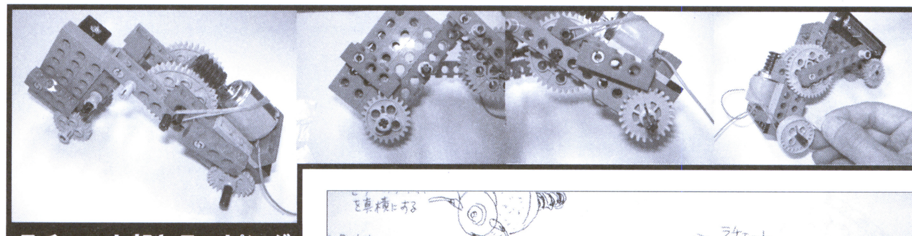




午前中に「しゃくとり虫」を作りました。うまく動かすコツは以下のとおりです。

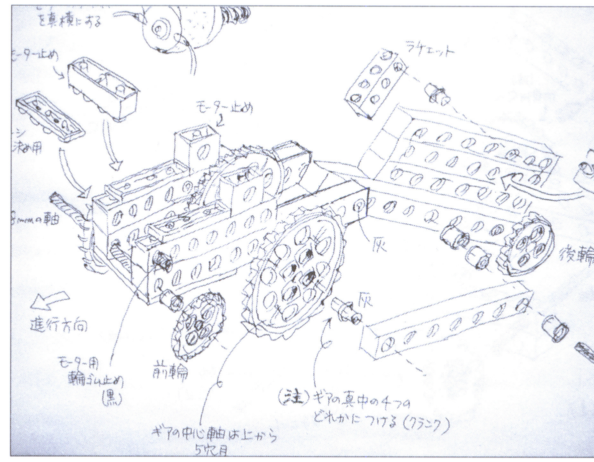
クランクで1動作するごとにラチェットで交互に前輪と後輪を押さえるので真ん中で折れる胴体がしゃくとり虫のようにくねくねと動きます。全体図は小さくなって見にくいと思いますが写真で確認してください。

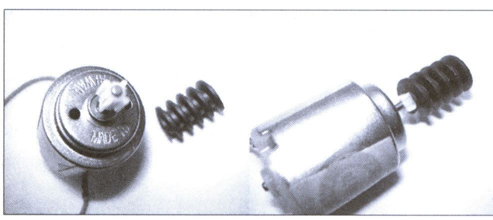
コツはモータを止める輪ゴムをあまりきつくしないこと，モータにビニールテープを巻いてウオームと40枚歯ギアとの噛みあわせを調整することです。ラチェットの位置を間違えないこと，前進側の車輪にしている25枚歯ギアにもビニールテープを巻いて床との摩擦を増やしてやることです。また，電池ボックスは両面テープで止めますが，あまりタップリつけるとはがすときが大変です。

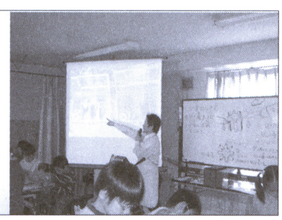


ラチェット部とテーピング









タミヤのピニオンギアがマブチモータの軸にピッタリ合うので，上の写真のように歯を1枚おきに削って4枚にし，十の形状にします。こうすればLEGOのウオームギアにはまります。LEGOは「ばらせる」という大前提がありますので，モータも輪ゴムで止めたりするのです。

LEGO以外の部品の使用，「たとえばタッピングビスで止める」という方法は，LEGOの工作規定に違反すると考えます。少し“こだわり”ですけどね！



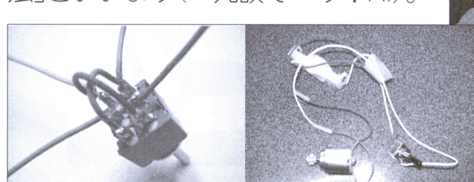
▲毎月遠くから本当にご苦労様です。これは楽しいお昼時間のひとコマです

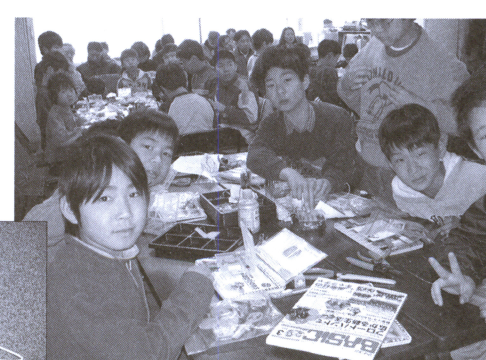
前進・後進切り替え回路をハンダ付け工作しました

モータの回転を逆転させるには，2回路2接点のスイッチを右の写真のように配線すれば簡単に仕上がります。配線をバッテンにするわけです。先月号でもバッテン配線がでてきましたが，バッテン配線はおもしろい電気回路ですね。

バッテンに配線して，真ん中に電池，上下どちらかにモータをつなげば簡単に出来上がります。つないでみて思ったのと逆に回ったら，電池でもモータでもどちらでもいいので＋と－を逆に接続すればOKです。

この方法を「いきあたりばったり法」といいます(←冗談でーす！Hi)。

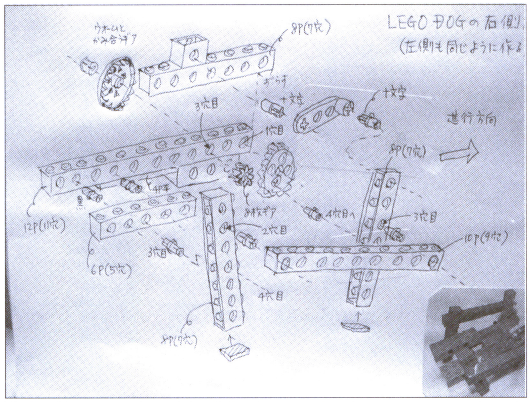




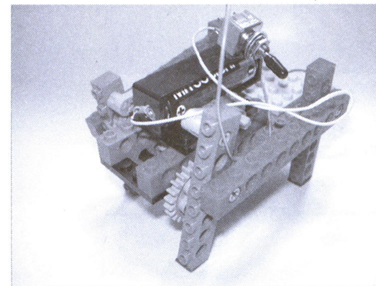
午後からは同じLEGOで4足歩行ブルドッグを作りました

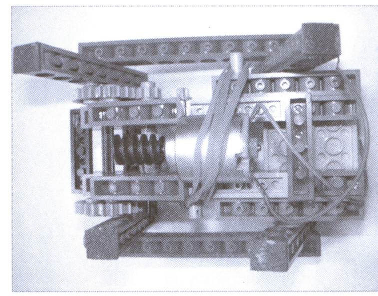
先月まではタミヤのメカドッグを作りましたが，これをLEGOで出来ないか，と考えたのが川森先生がチエを絞ったLEGOブルドッグです。これも同じモータとウオームギアで駆動します。歩行に必要なカムとリンク機構の勉強です。足の裏に滑り止めの両面テープを付けるのがコツです。LEGOはパーツをなくすと入手に手間がかかるので，細心の注意をはらって，なくさないように心がけましょう。

でんでんタウンロボットクラブでは，100円ショップで便利な仕切り付きの箱を見つけて部品入れに使いました。



※LEGOで作るブルドッグやしゃくとり虫の組み立て方を希望の場合は次ページのでんでんタウンマップ下に記載の「でんでんタウン電子工作教室」宛に住所，氏名，年齢を記入のうえ，ハガキで申し込んで下さい。詳しい資料（白黒）をお送りします。





次回活動日は4月13日です。楽しい競技大会です！

3月は2種類のLEGOを完成させるのが精一杯で，時間切れとなってしまいましたので，競技大会は4月13日にお預けとなりました。4月6日に特別メンテナンスのサービスデーを設けて，動くLEGOの最終チェックを行います。


ベーマガとは長ーいおつきあいやで

みんなの前にはいつもマイコンBASICマガジンが置かれています。

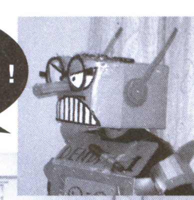
この本で勉強して，青春を謳歌して，将来これがきっかけでロボットやプログラムのプロになってほしいと願っています。

でんでんタウンロボットクラブでは8月までに2足歩行ロボットも闊歩させようと，今もくもくと計画を進行中です。

▼川森先生（左）と毎回裏方で大忙しのスタッフ達



ARISSコンタクトなど夢ある今後の活動



このロボットクラブの母体は，でんでんタウン電子工作教室です。時代とともに電子工作も変わっていきますが，完成した時のワクワク感は変わりません。工作教室では今後も次々と新しい電子工作ネタを見つけては挑戦を続けます。

秋には究極のロボットアームを搭載する国際宇宙ステーションとアマチュア無線で交信するARISSコンタクトに挑戦しようと頑張っています。

アマチュア無線クラブのコールサイン「JL3YIC」が聞こえていたら，よろしく！

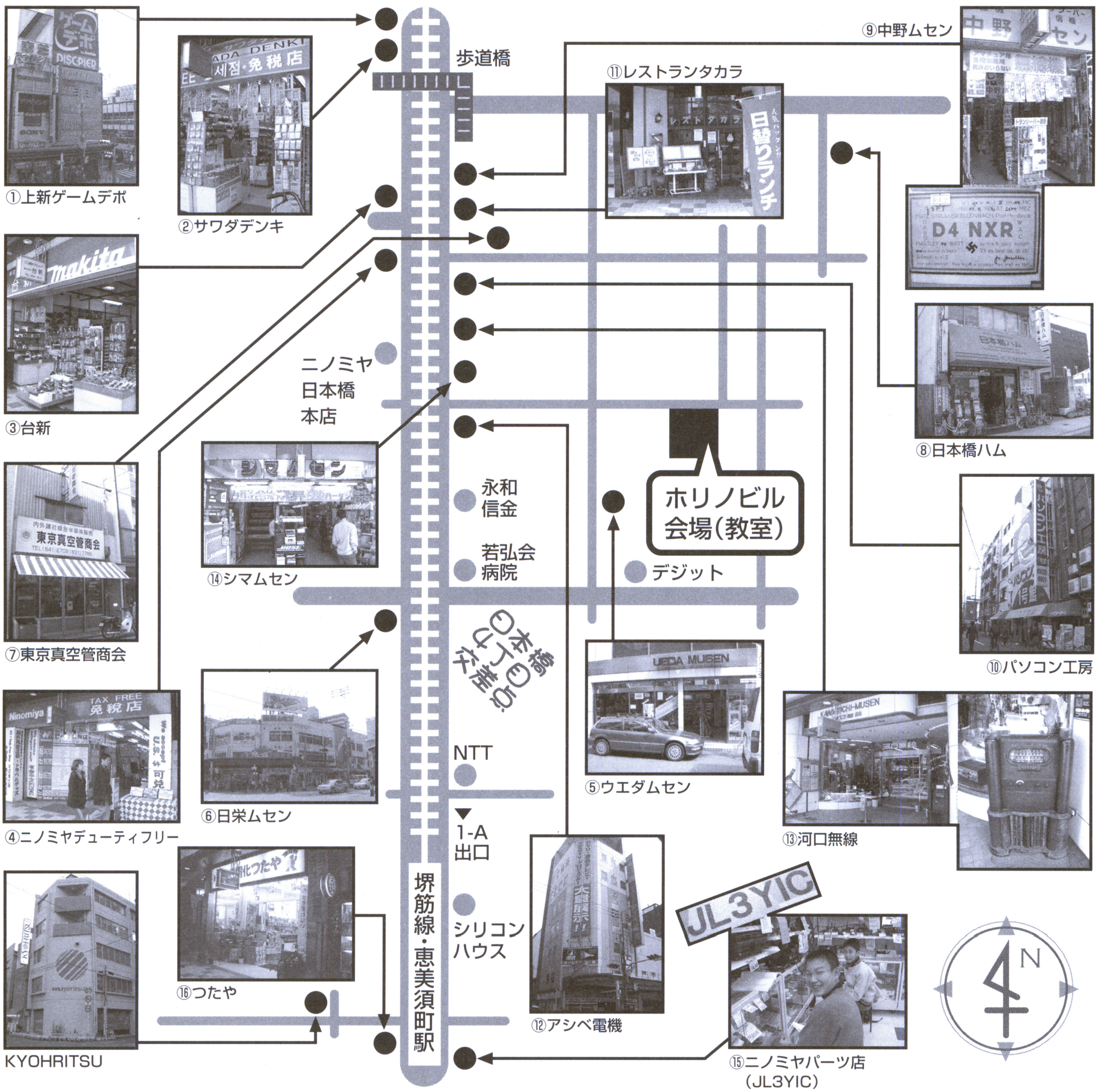
みなさん，挑戦を続けるでんでんタウンで是非お会いしましょう。ごきげんよう！



でんでんタウンロボットマップ

専門店のある街「でんでんタウン」マップ

今月は日本橋ならではの専門店を紹介します。①ゲームソフト専門：上新ゲームデポ　②免税専門：サワダデンキ　③電動工具専門：台新　④免税専門店：ニノミヤデューティフリー　⑤アマチュア無線専門：ウエダムセン　⑥アマチュア無線専門：日栄ムセン　⑦真空管専門：東京真空管商会　⑧アマチュア無線講習会専門：日本橋ハム　⑨アマチュア無線専門：中野ムセン　⑩組み立てパソコン専門：パソコン工房　⑪洋食専門：レストランタカラ　⑫プラズマ，プロジェクター専門：アシベ電機　⑬オーディオ専門：河口無線　⑭オーディオ専門：シマムセン　⑮パーツ，アマチュア無線専門：ニノミヤパーツ店　⑯200V家電専門：つたや

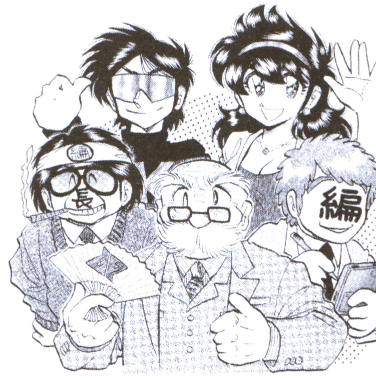


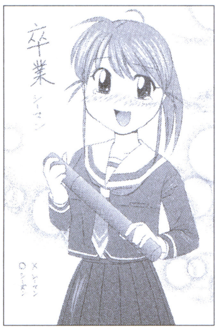
おもろい情報：東京真空管には１本数十万円の真空管が！　サワダデンキには中国語ぺらぺらのおねえさんが！　ウエダムセンには外国製のトランシーバーが！　中野ムセンには統合前のドイツの交信証（QSL）が！　河口無線には懐かしの大型ラジヲが！　ニノミヤパーツではJL3YIC局を運用できる！　KYOHRITSUには有名な味マップが！　などなど日本橋でんでんタウンはおもろい専門店がいっぱい！

行ってみよう日本橋へ！レッツGo！

〒556-0005 大阪市浪速区日本橋4-5-19 ホリノビル302 日本橋筋商店街振興組合　でんでんタウン電子工作教室　☎06-6641-6111

あくせすROOM





三重県四日市市・ぶんちょ

編「みなさんも，ベーマガを卒業できましたか？」



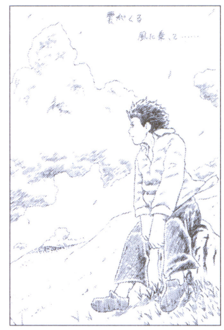
埼玉県所沢市・野沢プロト

Dr.D「先人の屍を乗り越えて，ビックになるのじゃ」



神奈川県平塚市・ら抹茶

編「ペグちゃんの頭突きは痛そうです」



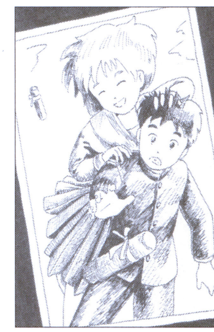
大阪府吹田市・ARAI

影「すわ，地震雲か!?」



岐阜市・極堂

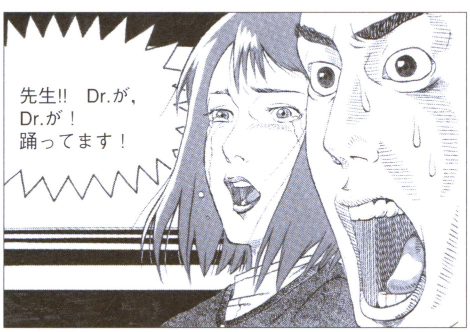
つぐ美「いまだに珍しいCG投稿です



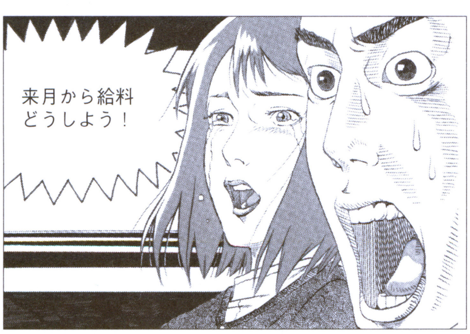
熊本県玉名郡・巨石人

編「卒業証書は大事にとっておこう！」

穴埋め問題



愛知県名古屋市・最馬場可



愛知県名古屋市・リストラ大王

お楽しみプレゼント当選者

2003年2月号
①神奈川県平塚市・牟田一大さん他9名　②愛媛県新居浜市・山田誠吾さん他19名　③愛知県名古屋市・古川純一さん他1名　④埼玉県比企郡・山崎敬一さん　⑤神奈川県相模原市・中村大亮さん

2003年3月号
①徳島県鳴門市・岩田直さん，東京都渋谷区・岩瀬裕子さん　②北海道浦河郡・亀井英俊さん　③秋田県本荘市・阿部穂高さん　④三重県度合郡・坂本光優さん

あくせすROOMへの投稿ありがとう

編「あくせすROOMは今月でおしまいです。今まで応援ありがとう！」

影「これからは，みんなの心の中に投稿コーナーを作って，切磋琢磨しよう」

つぐ美「いいこと言ってるのか，ただの変な人なのか，わかんない……」

Dr.D「読者の諸君らは，いずれワシらのことを忘れてしまうかもしれんが，ワシらは諸君のことを，ずっと覚えているぞ」

編集長「この21年間，いい勉強をさせてもらったね」

「マイコンBASICマガジン」休刊のお知らせ

平素は「マイコンBASICマガジン」をご愛読いただきありがとうございます。

さて，突然ではありますが「マイコンBASICマガジン」は，この5月号をもって休刊いたすことになりました。

1982年創刊以来，コンピュータが大好きな中・高・大学生を中心に多くの読者に愛読され続け，21年という永い歴史を築くことができましたのも，ひとえに読者・スポンサー様・関係者の皆様方の暖かいご支援の賜物と厚く感謝申し上げます。

振り返れば，コンピュータ業界で活躍されている方の中には，「マイコンBASICマガジン」を登竜門として現在に至っている方も多く，業界に果たしてきた役割はそれなりに大きかったのではないかと自負しています。

さりながら時代の変化にともない，小誌の社会的役割も終えたものと思慮し，永い歴史に幕をおろすことになりました。今後は，時代を先取りするような新しいスタイルで皆様とお会いしたいと思います。

ありがとうございました。

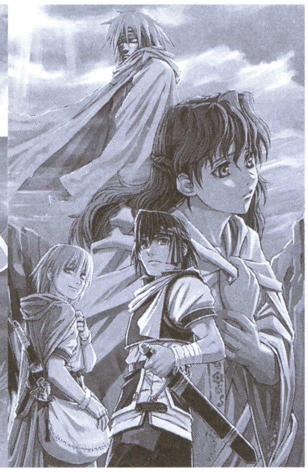
電波新聞社　マイコンBASICマガジン編集部

Falcom® NEWS EXPRESS

日本ファルコム株式会社
〒190-0012 東京都立川市曙町1-14-13
TEL 042-527-6501
FAX 042-528-2714

スタッフ募集中
採用担当BFN係
http://www.falcom.co.jp/

不朽の名作続々登場！
XP対応・お手頃新価格新パッケージで5／23一挙発売決定！

ファルコムXP定番キャンペーン実施決定！

*期間中，プレゼント対象製品を購入された方に，イースや白き魔女などの特製リストレスト（クリーナータイプ）をもれなくその場でプレゼント！詳細はファルコムHPで！

発売ラインナップ

■ 英雄伝説V海の檻歌
■ 英雄伝説IV朱紅い雫
■ 英雄伝説III白き魔女（特典：交響幻想曲白き魔女）
■ Zwei!!（特典：Zwei!!スーパーアレンジバージョン）
■イースI完全版
■イースII完全版

メディアカイトから廉価版が4/18発売！

PCゲーム初心者に最適！アクションが苦手な方のためにゲームバランスを調整した「イース1VE」。英雄伝説シリーズの原点である英雄伝説1をWindowsへ移植した「新英雄伝説」。ソーサリアンの入門編ともいえる「ソーサリアン・フォーエバー」の3タイトルが，お手頃価格で登場！PCゲームは難しいと思っている方は，ぜひこちらから試してみては？

■イース１ＶＥ　価格：3600円
■新英雄伝説　価格：2500円
■ソーサリアン・フォーエバー　価格：2000円

発売元：日本ファルコム株式会社
販売元：株式会社メディアカイト

全ての英伝ファンに捧ぐ！完全永久保存版「ガガーブトリロジー回想録」4月下旬発売！



白き魔女，朱紅い雫，海の檻歌，3つの世界で綴られる物語の世界観・キャラクター・ストーリーを細部まで紹介。さらに，次回新作の英雄伝説VIの最新情報までもが掲載され，総ページ数は何と776ページにも及ぶ，まさに究極の英伝ファンブックが遂に登場！ぜひ手元に置いてほしい一冊だ。

発行：新紀元社
予価：3700円

ファルコム携帯コンテンツ専用ページ「ファルコムモバイルステーション」ＯＰＥＮ！

ファルコムミュージックやゲーム，待ち受け画像などが携帯電話でゲットできます。「イース」などのファルコム代表作はもちろん，ファルコムキャラクターを使った携帯限定のゲームも！お使いの携帯電話でぜひアクセスしてみて！

★ ゲームの殿堂!!!「ブランディッシュ」「イースクロニクル」他
★ ｉゲーム大好き！「ザナドゥ」「イースIII」
★ ゲームキャラメロ100回
★メロディクリップ
★ RPG好き！「イース」
各サイトへのアクセス方法及び詳細は下記URLへアクセス！
http://www.falcom.co.jp/mobile/index.html

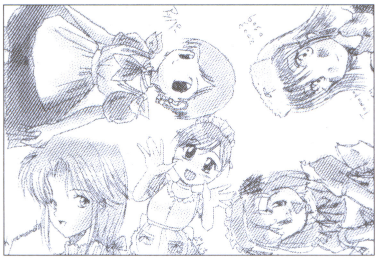
VM JAPANトレーディングカード・テレホンカード好評発売中！





「VM JAPAN」のグッズがホビージャパンより新登場！トレーディングカードは全36種類，そのうち18種類はメタリック加工。隠しキャラクターのリリアやダルク＝ファクトのカードもあり！テレホンカードはメリー・ルーがプリント。トレカ，テレカどちらも限定生産のためお早めに！お求めはファルコムオンラインショッピングコーナーで。

トレーディングカード　価格：4800円（1BOX）
テレホンカード　価格：1500円
http://www.falcom.co.jp/mailorder/



さようなら なにわ通信

タイトル冠文：大阪府・暁
タイトル・イラスト：中部地区の方々
本文：暁

PRESENTED BY マイコンソフト
http://www.micomsoft.co.jp
ADDRESS● 〒530-0005大阪市北区中之島3-2-4
朝日新聞ビル6F電波新聞社大阪本社内
マイコンソフト株式会社 なにわ通信係

みなさん，こんにちは。
今日はちょっと悲しいお知らせです。
長い間，皆様に親しまれてまいりましたベーシックマガジンは，本号をもってお休みをいただくこととなりました。このため，なにわ通信も今回が最終回となります。

新製品

3月10日に，新製品のモバイル型ビデオ・スキャンコンバーター「XMOV-1」が発売されました。本機はUSB電源で動作する，映像コンバーターで，パソコンのRGB映像を家庭用のテレビに手軽に表示する事ができます。価格は12,800円(税別)です。
ご自宅や，職場で1台いかがでしょうか?

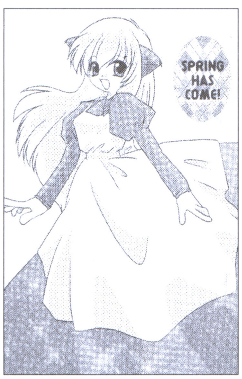
新製品2

以前よりご紹介しております，D端子セレクター「XSELECT-D4」も発売となります。
運がよければ，この本が発売されている頃には販売店様の棚に，製品が並んでいるかもしれません。詳しい発売日や仕様については弊社ホームページをご覧ください。
マイコンソフト社のホームページは3月下旬にリニューアル・オープンしました。
なにわ通信が終了した後も，製品情報については弊社ホームページにてご覧いただけます。

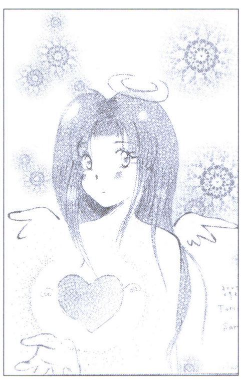
気が付くと

気が付くと，なにわ通信はベーマガの中でも，最も古いコーナーの一つとなっていました。初代の担当者「K村氏」から始まり，2代目の「Hiaro氏」そして3代目の私と，13年間続いて参りました。その間，多くの読者の皆様に支えられてきました。
最後になりましたが，ベーシックマガジンや，なにわ通信を愛読していただいた方，投稿していただいた方，編集でお世話になった方々に厚く御礼申し上げます。
本当にありがとうございました。
そして，さようなら。

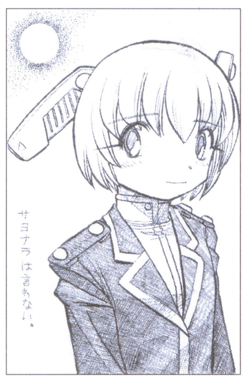
たくさんのおハガキ，ありがとうございました！



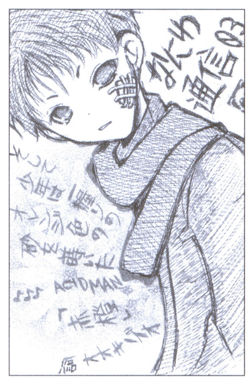
岐阜県・極堂
お久しぶりの投稿ですね。
ありがとうございました。
幸せいっぱいの極堂さんです。
赤ちゃんの予定はいつですか?



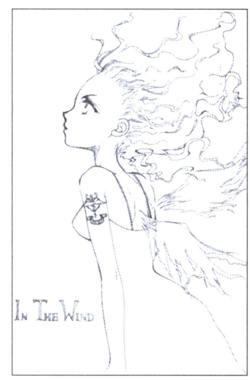
三重県・猿ともこ
天使さんでしょうか。ハートと翼が可愛らしいですね。長い間投稿していただきありがとうございました。



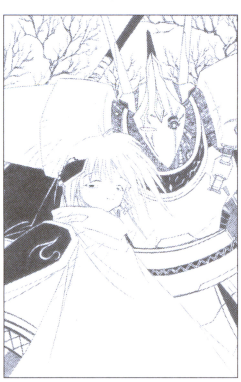
埼玉県・野沢プロト
耳にアンテナらしき物が付いておりますが，アンドロイドさんでしょうか。長い間投稿していただきありがとうございました



大阪府・藤井朱丸
ACIDMANがお気に入りのようですね。ボールペンの素朴なイラストをいつもありがとうございます。お勉強がんばってください。



東京都・天野真
ご無沙汰しております。腕のタトゥーがおしゃれな天使さんです。
投稿ありがとうございました。
お互いがんばりましょう。



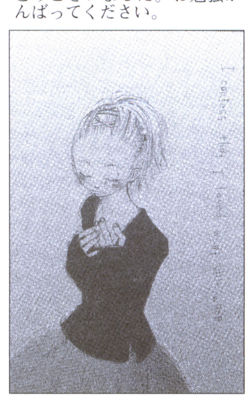
大阪府・ルストフレム様江
いつもパワフルな投稿で驚かされました。これからも，良い絵を描き続けてください。
投稿ありがとうございました。



京都府・陽東あひる
CGも上手いですよ！
長い間，投稿ありがとうございました。オート三輪を大事にしてあげてください。



三重県・ぶんちょ
三重県の生活もそろそろ慣れてきたでしょうか?
ぶんちょさんも投稿歴長いですよね。ありがとうございました。



北海道・青歌
高校時代の想い出を絵にした一枚でしょうか。大学へ行ってても描き続けてほしいです。ありがとうございました。



大阪府・Aす&さえき司
お2人のイラストを見ると，いつも心が和みました。ありがとうございました。赤ちゃんが元気に育つよう祈っております。

Q10　携帯・PHSを持っていますか？
①両方持っている ②携帯 ③PHS ④購入予定 ⑤予定なし

Q11　Q10で④の方のみ，どちらを購入する予定ですか？
①携帯 ②PHS ③両方 ④未定

Q12　使用中のプログラミング言語は？
※枠内に一つご記入ください

Q13　これから勉強したいプログラミング言語は？
※枠内に一つご記入ください

Q14　今月号の本誌で面白かった記事を三つ，下の表からお答えください

Q15　今月号の本誌でつまらなかった記事を三つ，下の表からお答えください

ご協力ありがとうございました。

■今月号の記事一覧

① 編集部の学校訪問記…6
② 僕たち私たちのホームページへようこそ!…14
③ あきば通が出来るまで…16
④ 創刊250号記念特集：コンピュータのルーツを探る旅…18
⑤ 未来のコンピュータ…28
⑥ Palm vs PocketPC プログラミング対決！…30
⑦ 今月の製品情報…36
⑧ 記録型パソコン用DVD技術を知る…38
⑨ マンガで学ぶ!コンピュータの世界!!…44
⑩ はじめようBASICプログラミング…48
⑪ Visual Basicアルゴリズム講座…52
⑫ デジタルの世界へようこそ…56
⑬ 特選投稿プログラム・コーナー…59
⑭ 今月のベスト・プログラマーは誰の手に！…101
⑮ Dr.Dのベスト・プログラムへの道…102
⑯ HSPdeネットワークゲーム！…104
⑰ .NETでゲームを作ろう！…108
⑱ パソコン認定試験対策講座…115
⑲ PIECEプログラミング…123
⑳ 続・ベーシックプログラミング教室［番外編］…127
㉑ でんでんタウンロボットクラブ奮戦記…131
㉒ あくせすROOM…135
㉓ SOFTWARE MAKER HOTINFORMATION…136
㉔ 表紙
㉕ OVER FLOW

キリトリ線

マイコンBASIC Magazine　プログラム投稿応募カード（ムック発行の可能性もあります）

言語名（音源名）					
プログラム名（曲名）					
氏　名		ペンネーム		年齢	歳
住　所	〒				
TEL	※連絡可能な時間帯		応募年月日	年　月　日	

キリトリ線

郵便はがき

お貼り下さい 50円切手を

141-8715

（受取人）

東京都品川区東五反田1−11−15

電波新聞社

月刊 マイコンBASICマガジン 編集部

5月号 行

（フリガナ）		年齢	性別
ご氏名		歳	男・女
□□□-□□□□		電話	
ご住所	都・道 府・県		

職業：小学生／中学生／高校生／高専生／短大生／大学生／専門学校生／予備校生／アルバイト／公務員／会社員／その他（　　）

本誌を 毎月・時々・初めて 読む　何人で読みますか（　）人

自分のホームページを　持っている・持つ予定・持っていない・予定なし

本誌のほかによく読む雑誌は？［　　］

あなたは今なにに興味を持っていますか？

キリトリ線

月刊マイコンBASICマガジン⑤ 愛読者カード

★OFコーナー宛 ♥あくせすROOM宛 ■質問・感想（本誌への掲載／可・不可）

ペンネーム：

■アンケート回答記入欄 ■ ※黒の筆記用具でわかりやすく，はっきりとご記入ください

Q1		Q2	〔メーカー名 〕
Q3		Q4		Q5	
Q6			
Q7			〔商品名 〕
Q8			
Q9			
Q10		Q11	
Q12 〔 〕		Q13 〔 〕	
Q14		Q15	

■アンケートにご協力ください。

※左の愛読者カード・記入欄に，以下の質問に該当する番号をご記入ください。複数回答の質問で，回答欄をすべて埋める必要はありません。

Q1　メインで使っているパソコンは？
①デスクトップ ②ノート ③持っていない

Q2　パソコンの種類は？（Q1で③以外の方）
①自作 ②ショップブランド ③メーカー製
※枠内にメーカー名も記入してください

Q3　メインで使っているOSは？（Q1で③以外の方）
①Windows 95 ②Windows 98 ③Windows Me ④Windows 2000 ⑤Windows NT ⑥Windows XP ⑦Macintosh ⑧UNIX ⑨Linux ⑩その他

Q4　パソコン歴は？
①1年未満 ②1～2年 ③3～5年 ④6～10年 ⑤11年以上 ⑥なし

Q5　インターネット歴は？
①1年未満 ②1～2年 ③3～5年 ④6～10年 ⑤11年以上 ⑥なし

Q6　パソコンの主な使用目的は？（複数回答可）
①プログラミング ②ゲーム ③インターネット ④メール ⑤ミュージック ⑥学習 ⑦CG ⑧その他

Q7　これから買いたいと思うコンピュータ機器は？（複数回答可）
①パソコン本体 ②プリンタ ③スキャナ ④デジタルカメラ ⑤その他
※わかれば，商品名も記入してください

Q8　持っているゲーム機は？（複数回答可）
①プレイステーション（PSone含） ②プレイステーション2 ③ドリームキャスト ④ニンテンドウ64 ⑤ゲームキューブ ⑥Xbox ⑦ゲームボーイ ⑧ゲームボーイカラー ⑨ゲームボーイアドバンス ⑩ワンダースワン ⑪ワンダースワンカラー ⑫その他

Q9　これから買いたいと思うゲーム機は？（複数回答可）
①プレイステーション（PSone含） ②プレイステーション2 ③ドリームキャスト ④ニンテンドウ64 ⑤ゲームキューブ ⑥Xbox ⑦ゲームボーイカラー ⑧ワンダースワンカラー ⑨ゲームボーイアドバンス

※裏面へつづく→

キリトリ線

■メモ欄…プログラム担当者・編集部へのご意見・ご感想など



「西風のラプソディー」に続く、ファルコム海外プロデュース作品第2弾！
「お前が天下一となれ！」
剣戟アクションが熱い！武侠RPGの決定版！
■絡み合う因縁と戦い、若き剣士が辿る宿命
■激しいアクション、新感覚・武功システム
■緻密な描写、息を呑む美しい情景
■心癒される大陸的サウンド
己の野望のために、俺の娘を捨てたのか……その点では俺とお前は似ている！
修煉
等級 10/10
経験 100005/0
蝕骨血刃
怒りによって、全身の気を爆発させる。簡単な技だが莫大な力を持つ。最高レベルになるととても強
：もし私の髪飾りを探し出せたら、戻ってあげる！
好評発売中！
ファルコムだから実現した、China英雄伝説！
武侠アクションRPG『月影のデスティニー』
for Windows® 98/2000/Me/XP
初回特典
サウンドトラック「月影のデスティニー」付き
月影のデスティニー
月影のDESTINY
©1997-2003 Season Software Ltd. ©2002-2003 Softop Co.,Ltd.
©2003 Nihon Falcom Corporation. All Rights Reserved.
http://www.falcom.co.jp/
■最新情報はファルコムホームページをご覧ください。 ■社員募集(新卒・既卒・中途)！詳しくはホームページ「会社案内・社員募集」をご覧ください。
Falcom®
NIHON FALCOM CORPORATION
〒190-0012 東京都立川市曙町1-14-13
☎042(527)6501 http://www.falcom.com/

第二十二巻第五号通巻二五一号 昭和五十七年十二月六日第三種郵便物認可 二〇〇三年四月二十五日印刷 二〇〇三年五月一日発行
電波新聞社 〒141-8715 東京都品川区東五反田一－十一－十五 電話〇三(三四四五)六一一一

定価550円 本体524円 雑誌18361-05 T1118361050554

2003

わかる！動かせる！
プログラムが組める雑誌

マイコンBASIC Magazine

5

コンピュータのルーツを探る旅「パソコン怒涛編」

Palm vs PocketPC プログラミング対決！

http://www.basicmagazine.net/

電波新聞社